OPTIMIZATION OF HYBRID-ELECTRIC PROPULSION SYSTEMS FOR SMALL REMOTELY-PILOTED AIRCRAFT

THESIS

Todd A. Rotramel
Captain, USAF

AFIT/GAE/ENY/11-M27

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GAE/ENY/11-M27

OPTIMIZATION OF HYBRID-ELECTRIC PROPULSION SYSTEMS FOR SMALL REMOTELY-PILOTED AIRCRAFT

THESIS

Presented to the Faculty

Department of Aeronautics and Astronautics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Aeronautical Engineering

Todd A. Rotramel

Captain, USAF

March 2011

AFIT/GAE/ENY/11-M27

OPTIMIZATION OF HYBRID-ELECTRIC PROPULSION SYSTEMS FOR SMALL REMOTELY-PILOTED AIRCRAFT

Todd A. Rotramel

Captain, USAF

Approved:

____________________________________ ________

Frederick G. Harmon, Lt Col, USAF (Chairman) Date

____________________________________ ________

Christopher M. Shearer, Lt Col, USAF (Member) Date

____________________________________ ________

Dr. Marc D. Polanka (Member) Date

AFIT/GAE/ENY/11-M27

Abstract

Small electric-powered remotely-piloted aircraft (RPA) used by today's warfighters for intelligence, surveillance, and reconnaissance (ISR) missions lack desired endurance and loiter times, while the acoustics and thermal signatures of those configured with internal combustion engines (ICE) may make them unpractical for low altitude ISR. Outfitting RPA with parallel hybrid-electric propulsion systems (H-EPS) would meet the military's needs by combining the advantages of both systems while reducing fuel consumption and environmental impacts. An analysis tool was created, using constrained static optimization, to size the H-EPS components. Based on the RPA's required power and velocity for the endurance phase, an electric motor (EM) can be designed or selected and matched with a commercial off-the-shelf (COTS) propeller for maximum efficiency. The ICE is then sized for the RPA's required power and velocity for the cruise phase.

Acknowledgments

I would like to start out by thanking my wife for her patience and understanding; no matter what I get myself into, and my boys for being my inspiration in life. Everything I do is because of them. I would like to thank Lt Col Harmon, for being a great mentor and advisor. This thesis is based on work he started and I could not have finished without his patience and guidance. I would like to thank my fellow students and friends: Colin Greiser, Isseyas Mengistu, and Matt Ripple. We spent many hours together designing and building the dynamometer test rig and testing the H-EPS components. Finally, I would like to thank Brian Crabtree, Daniel Ryan, Christopher Harkless and their supervisor, Jan LeValley, for the excellent service they provided at AFIT Model Fabrication Shop fabricating critical components; sometimes at the last minute.

-Todd Rotramel

Table of Contents

Page

List of Figures

Page

Page

List of Tables

List of Abbreviations

AFIT	Air Force Institute of Technology
AFRL	Air Force Research Laboratory
BDC	Brushed Direct Current
BLDC	Brushless Direct Current
CF	Cost Function
COTS	Commercial Off-the-Shelf
DC	Direct Current
EM	Electric Motor
EV	Electric Vehicle
GA	General Aviation
GR	Gear Ratio
H-EPS	Hybrid-Electric Propulsion System
HE-RPA	Hybrid-Electric Remotely Piloted Aircraft
HEV	Hybrid-Electric Vehicle
ICE	Internal Combustion Engine
ISR	Intelligence, Surveillance, and Reconnaissance
MATLAB	Matrix Laboratory
MEP	Mean Effective Pressure
R/C	Radio Control
RPA	Remotely-Piloted Aircraft
SFC	Specific Fuel Consumption
SI	International System of Units
UAS	Unmanned Aircraft System
UAV	Unmanned Air Vehicle
US	United States

Nomenclature

Symbol	Description (Units)
C_P	Coefficient of power
C_Q	Coefficient of torque
C_T	Coefficient of thrust
D	Propeller diameter (m)
F/A	Fuel-to-air ratio
I	Current (A)
I_{Stall}	Stall Current (A)
I_0	No-load current (A)
$I_{\eta m\_Max}$	Maximum electric motor efficient current (A)
J	Advance ratio
K_v	Electric motor speed constant
K_Q	Electric motor torque constant
MEP	Mean effective pressure (N/m^2)
$\dot{m}_f$	Fuel mass flow rate
n	Rotational speed (revolutions/s)
n_R	Number of crank revolutions per power stroke
N	Rotational speed (rpm)
P	Power (W)
P_E	Electric power (W)
P_P	Propeller power (W)
P_{Shaft}	Shaft power (W)
Q_{HV}	Heating value (J/kg)
Q	Torque (N-m)
Q_m	Electric motor shaft torque (N-m)
Q_{Stall}	Stall torque (N-m)
R_m	Resistance (Ω)
SFC	Specific fuel consumption (kg/Wh)
T_p	Propeller thrust (N)

V	Voltage (V)
V_∞	Freestream velocity (m/s)
V_m	Internal motor voltage (V)
Λ	Delivery ratio
η_m	Electric motor efficiency
η_{m_Max}	Maximum electric motor efficiency
η_f	Fuel conversion efficiency
η_p	Propeller efficiency
η_{tr}	Trapping efficiency
η_v	Volumetric efficiency
ρ_∞	Density (kg/m^3)
ω_m	Motor rotational speed (rad/s)
ω_0	No-load speed (rad/s)
$\omega_{\eta m\_Max}$	Maximum electric motor efficiency rotational speed (rad/s)

OPTIMIZATION OF HYBRID-ELECTRIC PROPULSION SYSTEMS FOR SMALL REMOTELY-PILOTED AIRCRAFT

I. Introduction

1. Background and Motivation

Although the name may have changed at least a dozen times, the concept remains the same; develop an aircraft that can be flow without an onboard pilot. Today's more modern remotely-piloted aircraft (RPA) can be traced back to the visions of Nikola Tesla. In the 1890's, Tesla wrote an article that went unpublished in which he claimed he could invent a remotely-controlled aircraft [1]. Less than 15 years after Orville Wright made his historic first flight in December, 1903, the United States military flew its first unmanned air vehicle (UAV). The 'Kettering Bug" was truly unmanned. A barometer was used to reach a preset altitude which then would transition to a gyroscope for control. After a predetermined number of propeller rotations, the engine ignition would short-circuit, the wings would fall off, and the torpedo body would fall to the target [1]. RPA of today are far more complicated. Unlike the one-time-use "Kettering Bug", most are meant to be recovered and reused and may or may not involve a huge logistics footprint in order remove the pilot from missions coined as "dull, dirty, or dangerous".

Current RPA are used by many nations for various applications including military, border patrol, agriculture, wildlife monitoring, and fire patrol. According to *AIAA 2009 Worldwide UAV Round*, 45 countries manufacture hundreds of RPA, with the US responsible for a sizable bulk [2]. These RPA come in all shapes and sizes; ranging from the small, electric-powered, hand-launched RQ-11 Raven to the large, fully-autonomous, fuel-powered, RQ-4 Global Hawk. Electric-Powered RPA are quiet and stealthy, making them suitable for ISR missions, but lack the endurance demands of the US military. RPA powered by internal combustion engines (ICE) can keep up with the endurance demands, but lack the stealthy aspect at low altitudes, as well as, require greater logistic support. They are less efficient and therefore more costly to operate. A hybrid propulsion system would combine the best of both aspects. The concept of using a hybrid-electric propulsion system (H-EPS) for RPA was proposed by Harmon and is the basis for this thesis [3].

2. Problem Statement

Currently there are no tools available to match components of a hybrid propulsion system for small RPA in order to maximize efficiency.

3. Research Objective

This research focused on optimizing and matching the specific components of a propeller-based parallel hybrid-electric propulsion system based on power requirements and performance speeds for a small RPA. The four main objectives were to:

1. Verify electric motor (EM) manufacturer data by comparing to analytical and experimental data
2. Select and optimize specific components for the H-EPS
3. Incorporate experimental propeller data to assist with matching the EM and propeller for endurance
4. Determine optimized gear ratio for non-inline H-EPS

4. Research Scope

The optimization tool created by this effort is based on optimizing the components of a H-EPS for a RPA design proposed by Harmon [3] and later refined by Hiserote [4]. The aircraft design was not verified nor were any of the results. Although the H-EPS could be used throughout the RPA flight envelope, this thesis will evaluate only the cruise and endurance portions of a typical RPA flight profile to size the various components. Controlling the components in the H-EPS is a major endeavor and was also not evaluated in this effort. Information on the specific control logic is discussed by Greiser [5].

5. Methodology

The author started with a COTS propeller and implemented standard EM and propeller performance and design equations in a constrained static optimization formulation. This was done to maximize the efficiency of the EM and propeller combination for the endurance phase of flight. By setting the EM efficiency as the cost function and allowing the EM parameters to vary, the author was able to design the most efficient EM, predict the efficiency of an existing EM, or determine the optimal gear ratio for the existing EM and propeller combination. With the selection of the EM and propeller, the efficiency of the ICE was determined for cruise phase as well as the option for recharging the batteries for the EM.

6. Thesis Overview

Chapter I of this thesis provides an introduction to the thesis and relevant background information. Chapter II is a review of the particular components of the H-

EPS. Chapter III discusses the author's methodology. Chapter IV includes analysis and results. Chapter V discusses these results and recommendations for future work.

II. Literature Review

1. Chapter Overview

The concept of using hybrid-electric propulsion is not new. It has been around for quite some time, but due to rising gas prices and concerns for the environment, it has only recently been brought to the forefront. Today, this hybrid technology has been applied to cars, trains, busses and even boats. With the advancements of electric batteries, only now has the aviation industry truly been able to consider adapting what the automotive industry has proven. Adapting hybrid systems in aircraft could produce the same benefits seen by the automotive industry, as well as, increasing the capabilities of the military's RPA for ISR missions.

This chapter will briefly discuss the two main types of H-EPS systems as well as their background and applications. After which, the main system components will be discussed and analyzed in detail.

2. Hybrid-Electric Propulsion

Hybrid Electric Vehicles (HEV) are motivated by battery range limitations for electric vehicles (EV). By definition, a H-EPS contains two or more power sources, acting together or independently, of which one is an EM. Although, typically the other power source is an ICE, it could also be another power source such as a fuel cell. For the purpose of this thesis, the H-EPS is understood to contain an ICE, unless otherwise stated.

H-EPS combine the efficiency and clean power of an EM with the extended range of an ICE. The outcome is increased fuel efficiency and decreased emissions compared to conventional ICE powered vehicles; increased range compared to EVs, and overall improved vehicle performance. However, HEVs come with increased complexity and power system design. In general, H-EPS are either configured in a series or parallel configuration.

2.1. Configurations

In a series hybrid, as shown in Figure 1, the power from the ICE is transmitted electrically through a generator. The ICE and generator combination can either provide electrical power directly to the EM or to the battery which supplies power to the EM. In this configuration, the EM is the primary power source supplying the torque required to propel the vehicle. Because the ICE is not directly connected to the drive shaft, it can operate at its optimum torque and speed range; therefore decreasing fuel consumption

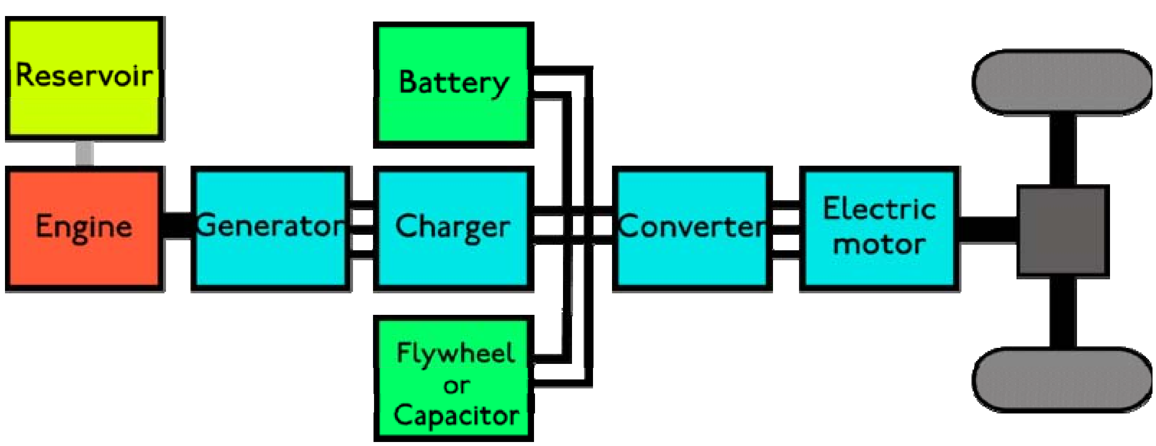


Figure 1: Series Hybrid-Electric Configuration [6]

and increasing efficiency of the ICE. The downside to the series configuration is that each power conversion process adds inefficiencies, complexity, and weight. Also, the EM has to be sized to provide the required power throughout the vehicles operational envelope which also adds weight.

A parallel hybrid system combines an ICE connected in parallel with an EM as shown in Figure 2. The ICE is mechanically connected to the drive train and therefore can directly supply the mechanical power. The EM is added in parallel so that it can either drive the system independently or supplement the ICE with torque for additional power requirements. Because the ICE does not have to produce all the required power, it can be downsized; therefore, saving weight. Even though the EM can be the primary power source, it typically is not for the entire envelope and therefore the EM torque required is lower than for the series hybrid. When the ICE is operating as the primary power source, the EM can be used as a generator to recharge the batteries or provide onboard electric power by absorbing the excess power produced when the ICE's output power is greater than the requirement to propel the vehicle.

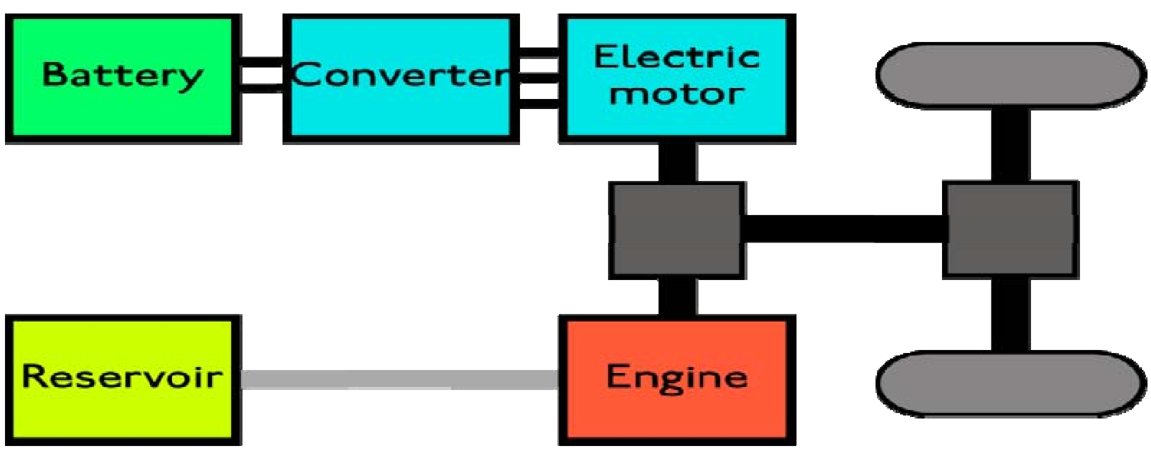


Figure 2: Parallel Hybrid-Electric Configuration [7]

The main concept behind parallel hybrid system is additive torque. As discussed, the torque of multiple power sources, operating at the same speed, will add together to increase the output power. The simplest demonstration of this concept is the tandem bicycle. Although powered by humans and not machines, the idea is the same. The individual torque, or effort, provided by each rider is less than the torque that a single rider would have to provide to operate the bicycle alone.

2.2. Applications

The first HEVs were built it the late 1890s and early 1900s. There seems to be some debate over who built the first one. In 1899, two hybrid vehicles were shown at the Paris Salon. One was a parallel hybrid, built by Henri Pieper of Belgium, in which a small ICE was coupled with an EM. The engine was the main power source and would charge the batteries when the vehicle was stopped or coasting to a stop. The EM would provide additional power when the demand was greater than ICE could provide.[8]. In 1909, the US Patent Office awarded Pieper the first ever HEV patent for his invention entitled “Mixed Drive for Autovehicles” [9].

The second HEV at the 1899 Paris Salon was a series hybrid built by Vendovelli and Priestly of France. It was a tricycle, with the back wheels driven by independent EMs, which pulled a small trailer mounted with an ICE and generator. Without the trailer, the tricycle was an EV, but the added trailer extended the vehicles range by utilizing the generator to charge the tricycle’s batteries [8].

Also around the same time, Ferdinand Porsche developed the *Lohner-Porsche Mixte Hybrid* [10]. He modified his previous *Lohner-Porsche*; an all electric vehicle that contained EMs in the wheel hubs. By adding an ICE and generator, in a series hybrid configuration, Porsche was able to charge the batteries of his electric vehicle and therefore creating the first production HEV [11].

Much like the first series H-EPS of Vendovelli, Priestly, and Porsche, locomotives are a good example of perhaps the oldest simple series HEV still around today. They use a diesel engine to drive a generator at constant speed. Power from the generator is fed directly to the electric motors in the wheels which propels the train down the track [12]. Some city busses and large dump trucks, such as the Caterpillar 797, use similar setups where the weight penalty is compensated by the high power efficiency.

HEVs did not become widely available to the general public until Japan introduced the Toyota *Prius* in 1997 and the Honda *Insight* in 1999 [13]. Although the *Prius* was initially only available in Japan, these vehicles were available throughout the world within a few short years. Since then, almost every automobile manufacturer has developed its version of a HEV.

In 2008, a group from the University of Padova, Italy designed a surface-mounted permanent magnet motor for use in a hybrid-electric propulsion system of a catamaran [14] . The series hybrid configuration utilized an ICE/generator combination to supply power to a submerged EM connected to the boat's propeller. From their analysis, the EM "appears to be well suited" due to the natural cooling provided by the water.

As stated earlier, electric hybridization of aircraft is only recently becoming a reality. A solution for general aviation (GA) had been developed by the German aircraft manufacturer, Flight Design. Flight Design coupled a 115-hp (85.8 kW) Rotax 914 airplane ICE with a 40-hp (29.8 kW) EM, via a belt, in a parallel hybrid configuration, as shown in Figure 3. The system is able to run for five minutes at maximum power, for

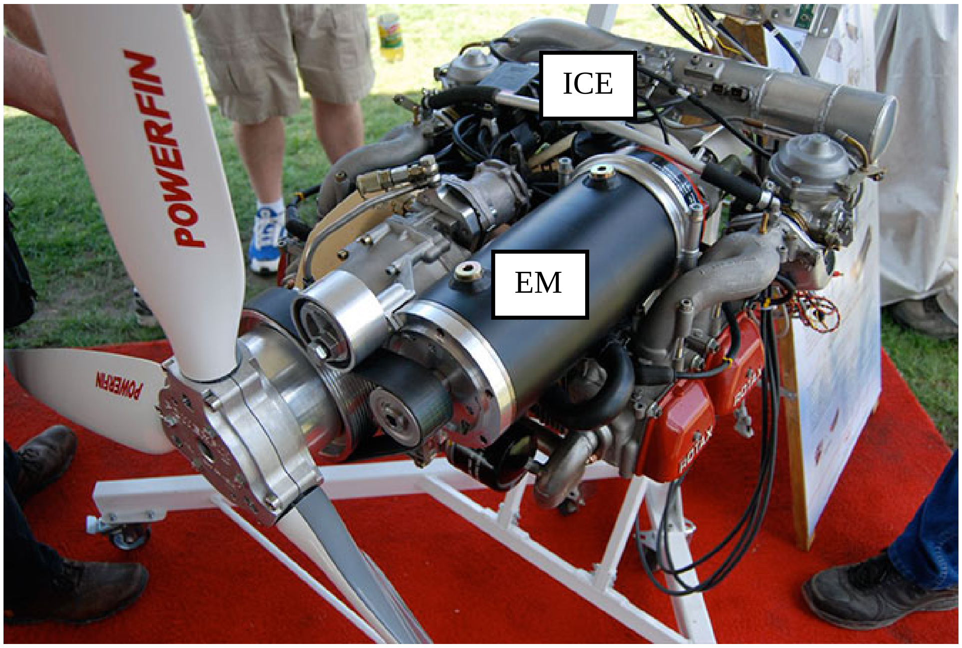


Figure 3: Flight Design's Hybrid-Electric Propulsion System [16]

take-off and climb, producing the equivalent power of the 160-hp (119.3 kW) ICE it is intended to replace. The aircraft can then cruise with the lighter, more efficiently sized ICE. Also, in true parallel hybrid fashion, the EM can power the aircraft, for a short amount of time, in the event of an emergency, or it can be used as a generator to recharge the batteries for future use.

A more appropriate design, shown in Figure 4, was tested by Richard Glassock of Australia. Glassock's idea was to downsize the 25 cc 4-stroke engine on Insitu's *ScanEagle*, pictured in Figure 5, and replace it with a 10cc 2-stroke engine, a rightly sized EM, and a Garvon 20"x6" propeller [15]. Like Flight Design's concept, the EM would provide the additional torque required at higher speeds to rotate the propeller. As shown in Section 3.2, the point at which the torque curve of the propulsion system intersects the torque curve of the propeller indicates the maximum speed for the specified conditions and therefore the maximum thrust available from the propeller.

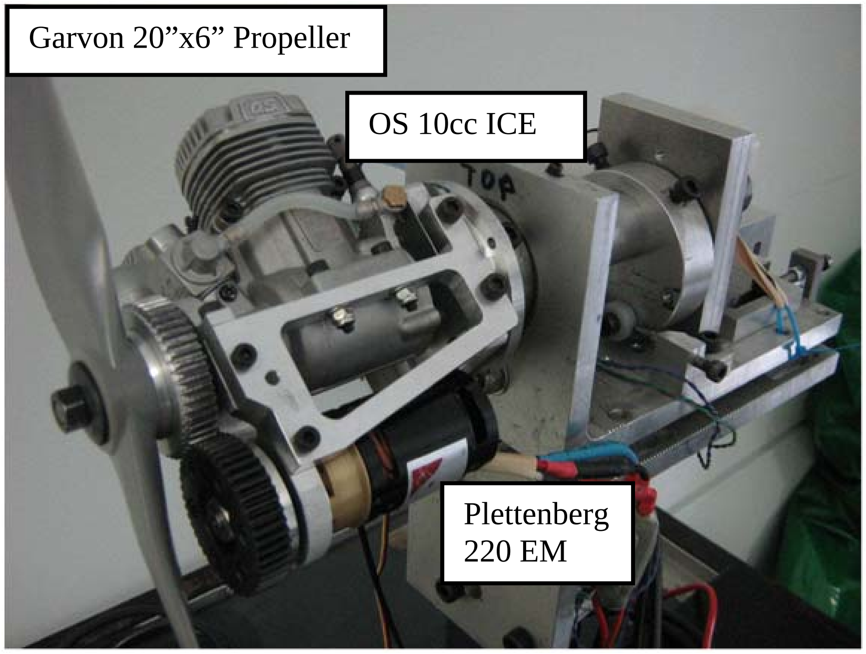


Figure 4: Glassock's Hybrid-Electric Prototype [15]

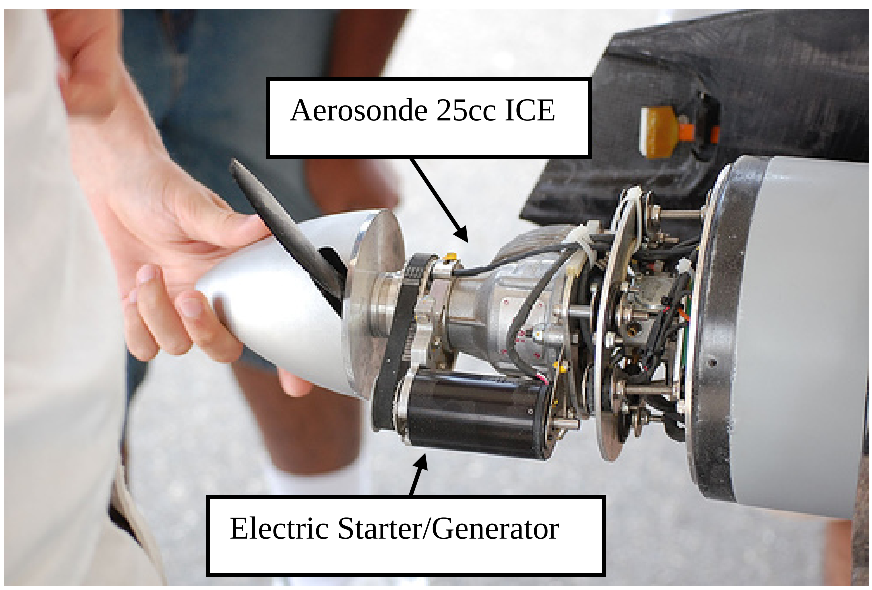


Figure 5: ScanEagle Propulsion System

Figure 6 shows the experimental results of Glassock's work. The figure on the left shows the load curves of the propeller and the 25cc engine intersect at around 6000 rpm, while the torque curve of the 10cc engine never intersects the propeller torque curve. The figure on the right shows that the additive torque curve of the 10cc engine and the EM due in fact intersect the propeller torque curve at about 7000 rpm for the static operating point (take off) and about 8000 rpm for the translational (cruise) operating point.

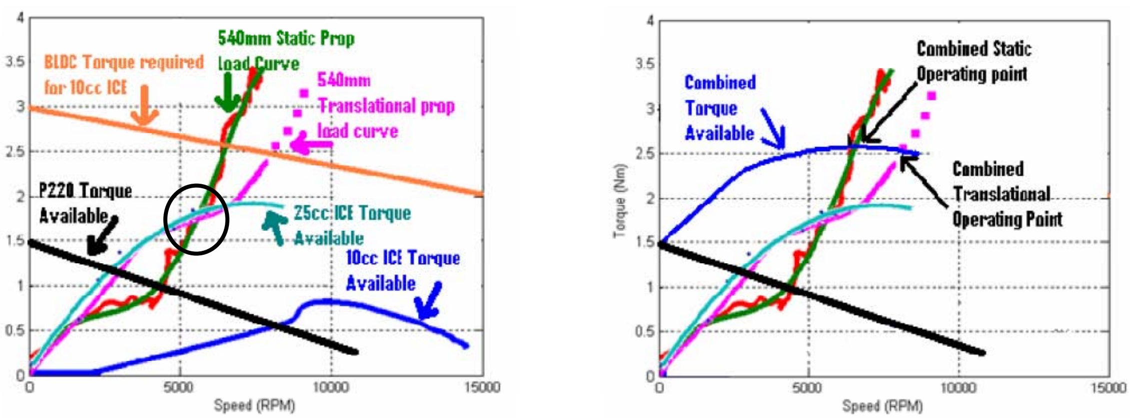


Figure 6: Glassock's Experimental Load Curves [17]

There has also been a great deal of research done on using fuel cells instead of internal combustion engines to power RPA and other small aircraft; hybrid or not [18][19][20][21]. Grant, et al, designed and is currently flight testing a fuel cell powered UAV with an estimated endurance time of 28 hours [22]. In 2008, Boeing flew a manned airplane powered by hydrogen fuel cells. The modified Diamond Aircraft *Dimona* motor glider was able to climb to an altitude of 3300 ft (1 m) using a combination of fuel cell and battery power. The aircraft was then flown for 20 minutes at 62 mph (100 km/hr) on fuel cell power alone [23].

3. Hybrid System Components

The three main components that make up the proposed H-EPS are the ICE, EM, and the propeller. Other important components are the batteries and the coupling system which could be a clutch, one-way bearing or a planetary gear system, and the gear ratio between EM and the propeller shaft. Each of the three main components must be analyzed in detail, but due to the scope of this project only the EM, with and without a gear ratio, and the propeller will be used to formulate a feasible static optimization problem. Only the operational speed range of the ICE will be considered.

3.1. Electric Motors

The main purpose behind this design is to fly low and quiet for an extended amount of time. This is accomplished by the EM and for this reason, the EM and batteries are the most important components of the system. Electric motors convert electrical power to mechanical power. They essentially use magnets to create motion. All brushed direct current (BDC) motors are made of the same basic components: a stator, rotor, brushes and a commutator. In a BDC motor, current is supplied to the commutator through the brushes to form a circuit between the electrical source and the motor's armature coil windings as shown in Figure 7. This produces an electromagnet which functions as the rotor. The shell of the motor is lined with permanent magnets of opposite polarity to form the stator. Due to the attraction and repulsion of the electromagnetic coil to the permanent magnets, the armature rotates. As the armature rotates, the commutator switches the polarity of the electromagnet and the armature continues to rotate.

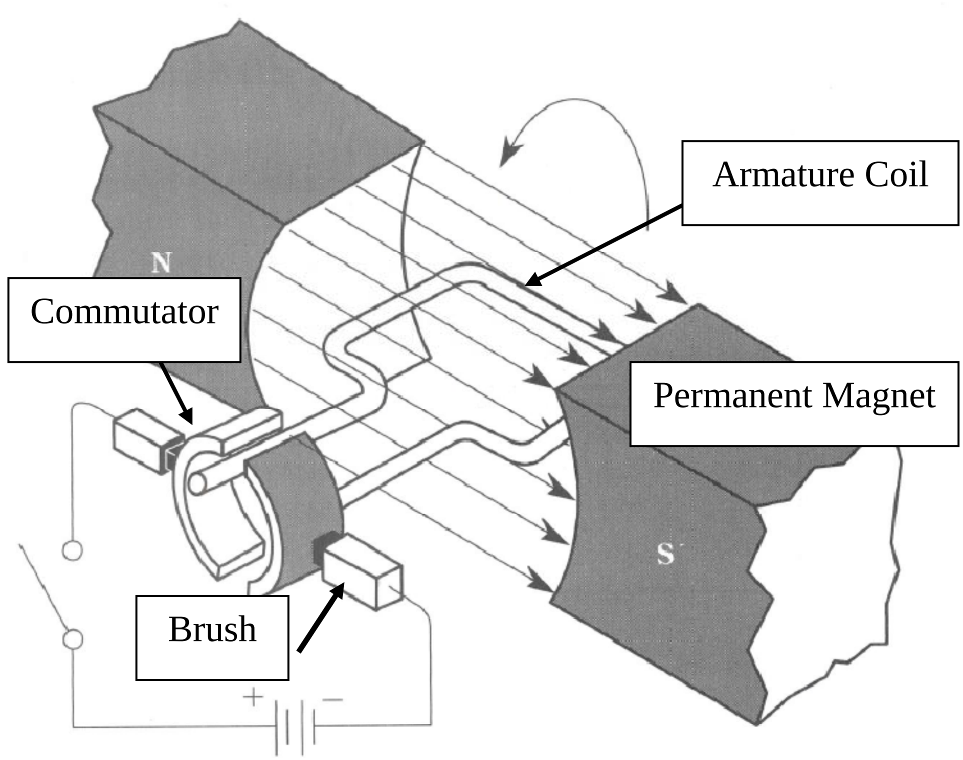


Figure 7: Brushed Direct Current Motor Diagram [24]

In a brushless DC (BLDC) motor, the electromagnets are stationary and the permanent magnets are on the spinning portion of the motor. Because the electromagnet is not rotating, a controller is required to switch the polarity of the circuit. Brushless motors are typically more expensive to build and more difficult to control, but with fewer moving parts, they are more reliable and efficient.

BLDC motors may either be inrunners or outrunners. The difference is the placement of the permanent magnets and electromagnets. An inrunner BLDC motor is similar to the brushed DC motor except the permanent magnets and electromagnets are in opposite positions. The permanent magnets are mounted directly on a spinning rotor and the stator windings of the electromagnet are attached to the motor housing. Figure 8 shows the electromagnetic stator of an inrunner BLDC motor. In an outrunner BLDC

Figure 8: Inrunner BLDC Motor Housing [25]

motor, the electromagnet is attached to the stator in the middle of the motor and the permanent magnets spin on a rotor that surrounds the stator. As shown in Figure 9, the stator of the outrunner BLDC is the motor housing.

Figure 9: Outrunner BLDC Motor

Although there are different types of DC motors, their basic operation is the same. Figure 10 represents an equivalent circuit for a DC electric motor as discussed by Drela [26], [27] and Ludstrom [28]. An EM is not 100 percent efficient. The mechanical power produced by the EM does not equal the electrical power supplied to the EM. This inefficiency is due to the EM's internal resistance (R_m) and its no-load current (I_0). As the name implies, the no-load current is the current drawn by the motor when there is no load on the motor shaft. This is assumed to be constant. As the voltage to the EM is increased, the rotational speed increases, but the current remains constant. Another important parameter of an EM is the motor speed constant (K_v). This is how EM's are rated. These parameters are usually supplied by the manufacturer but, if not, they can be measured. If the shaft of an EM is rotated at a known rotational speed, by an external device, and the output voltage of the EM is measured, the K_v can be calculated by simply dividing the speed by the measured voltage. That speed would be the no-load speed for that specific voltage. The parameter K_v is typically presented with units of rpm/volt, but

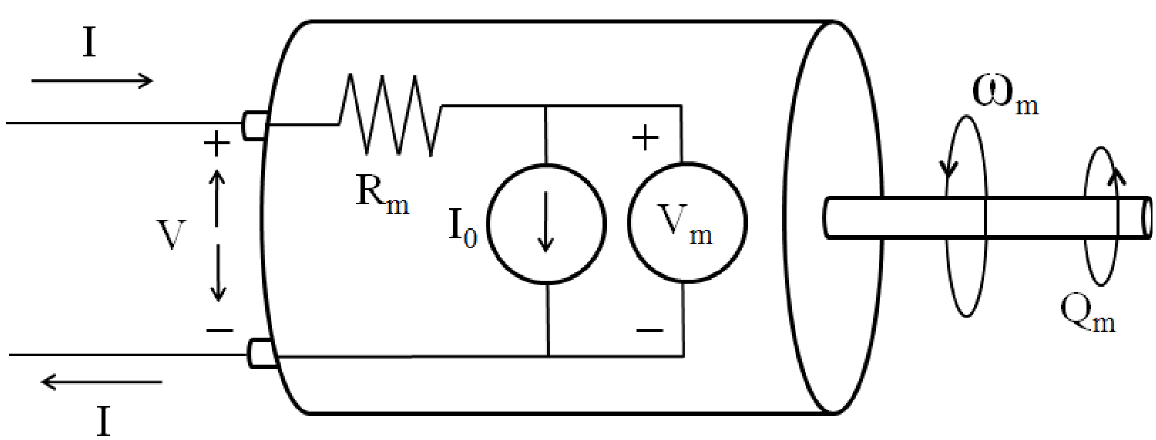


Figure 10: DC Electric Motor Equivalent Circuit

could also be seen as radians/sec/volt. From here, the rest of the circuit can be evaluated.

As depicted in Figure 10 the motor's internal back-EMF voltage (V_m) is the difference between the supplied voltage (V) and the product of the applied current (I) and R_m.

$$V_m = V - IR_m \tag{1}$$

The shaft rotation speed is the product of V_m and K_V. For the purpose of this thesis, ω represents rotational speeds with units of radians per sec (rad/s), N represents rotational speeds with units of revolutions per minute (rpm), and n represents rotational speeds with units of revolutions per second (rps).

$$\omega_m = (V - IR_m)K_v = V_m K_v \tag{2}$$

Just like K_V relates speed and voltage, there is a constant that relates torque (Q_m) to I called the motor torque constant (K_Q). This accounts for the inefficiency of the EM due to I_0. The English units are oz-in/A and the SI units are N-m/A.

$$K_Q = \frac{Q_m}{I - I_0} \tag{3}$$

If K_V is expressed in radians/sec/V and K_Q is expressed in N-m/A, then they are equal to the inverse of each other.

$$K_Q = \frac{1}{K_V} \text{ (In SI units)} \tag{4}$$

Therefore,

$$Q_m = \frac{I - I_0}{K_v} \text{ (In SI units)} \tag{5}$$

From the above equations and relationships, the EM efficiency can be determined. The electrical power supplied to the EM (P_E) is equal to the product of the supplied voltage and current and the output shaft power (P_{Shaft}) is equal to the product of shaft rotational speed and torque.

$$P_E = IV \tag{6}$$

$$P_{Shaft} = Q_m \omega_m = (I - I_0)(V - IR_m) \tag{7}$$

Solving Equation 2 for V and substituting into Equation 6:

$$P_E = I^2 R_m + I \frac{\omega_m}{Kv} \tag{8}$$

Solving Equation 5 for I and substituting into the right side of Equation 8:

$$P_E = I^2 R_m + \frac{I_0 \omega_m}{Kv} + Q_m \omega_m \tag{9}$$

Substituting Equation 7 into Equation 9 shows the efficiency of an EM due to I_0 and R_m.

$$P_{Shaft} = P_E - I^2 R_m + \frac{I_0 \omega_m}{Kv} \tag{10}$$

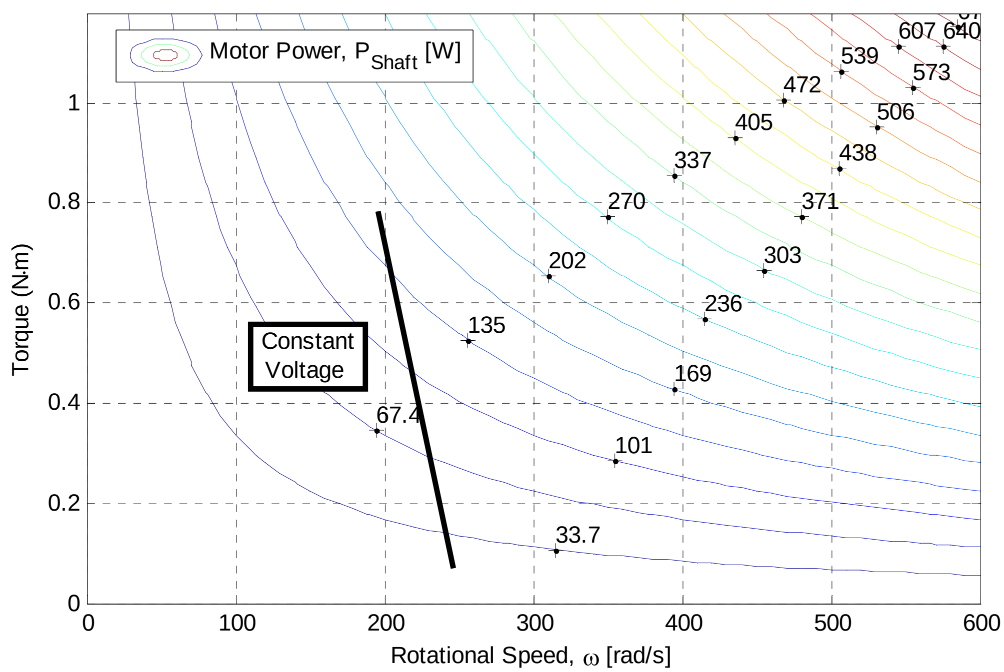


Figure 11 shows the power map of an EM supplied with a voltage range of up to 24 volts. The maximum speed occurs at the maximum voltage. As shown with Equation 7, as either torque or speed is increased, so is the output shaft power. Since both torque and speed are functions of current, if the voltage is held constant and the torque is increased, the current will increase (Equation 5), but the speed will decrease (Equation 2). This is shown by the bold black line in

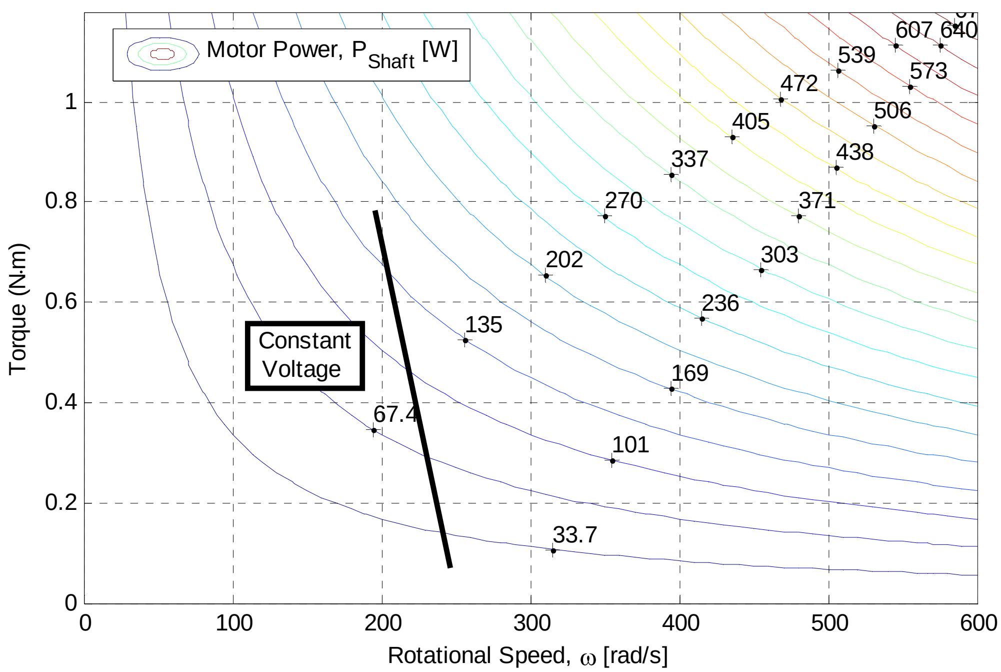


Figure 11. In the figure, the shaft power has increased, but the power may increase or decrease depending on the amount of current (Equation 10). This will be discussed in more detail later.

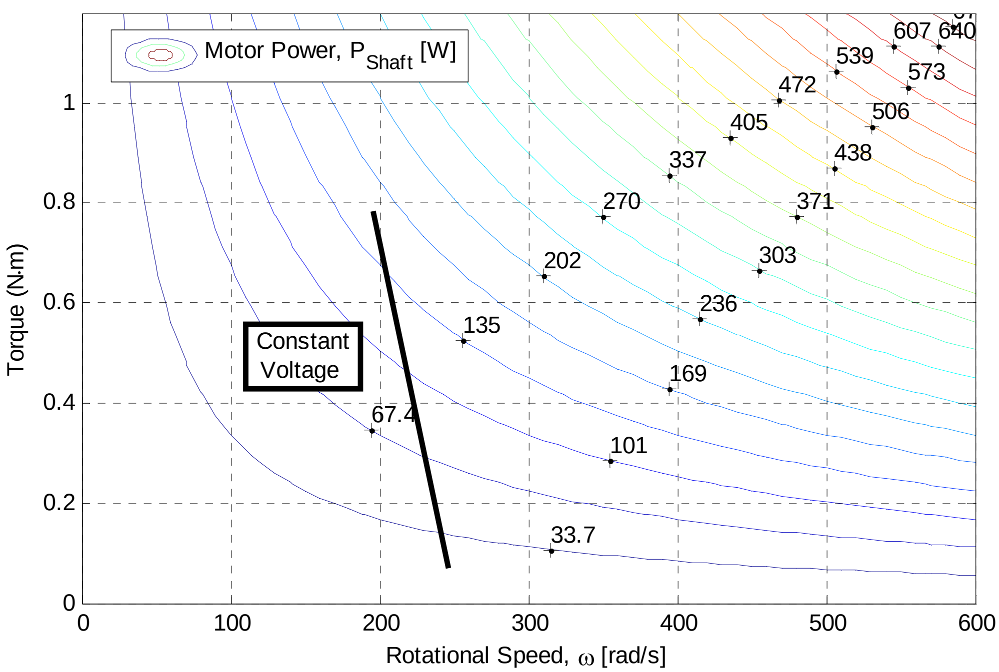


Figure 11: Electric Motor Power Map

For optimum efficiency, the EM should be operated such that the power lost due to R_m (ohmic heating) is balanced with the power lost due to I_0. Ohmic heating is the process by which the passage of an electric current through a conductor releases heat [29]. The efficiency of the EM (η_m) is equal the output power divided by the input power.

$$\eta_m = \frac{P_{Shaft}}{P_E} = \frac{(I - I_0)(V - IR_m)}{IV}\left[1 - \frac{I_0}{I}\right]\left[1 - \frac{IR_m}{V}\right] \tag{11}$$

Figure 12 shows the efficiency map of an EM supplied with a constant 24 Volts. As stated above, an increase in current (torque) will cause a reduction in speed when running at a constant voltage. It is unclear by looking at an efficiency map whether or not an increase in current will cause an increase or decrease in efficiency. Much the same as with power, it depends on the amount of current as shown in Equation 11. A better approach is to plot rotational speed, shaft power, and efficiency as function of torque or

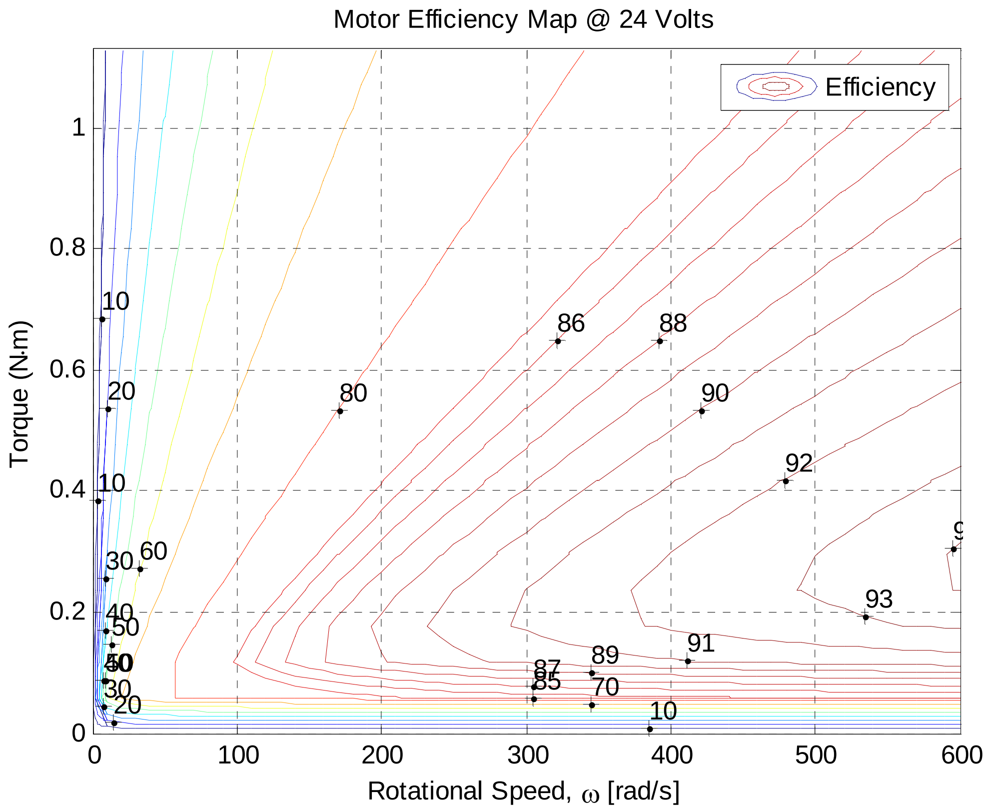


Figure 12: Electric Motor Efficiency Map

current as shown in Figure 13. Figure 13 clearly shows that as torque increases, rotational speed decreases and both power and efficiency increase and then decrease.

When the EM is operating at no-load ($Q_m = 0$), the rotational speed is the no-load speed (ω_0). The EM is stalled at the point when the rotational speed goes to zero. The torque at this speed is the stall torque, (Q_{Stall}) and the current at this stalled condition is the current required to start the EM, (I_{Stall}) [30].

$$\omega_0 = V * Kv \tag{12}$$

$$Q_{Stall} = \frac{V - I_0 * R_m}{K_v * R_m} \tag{13}$$

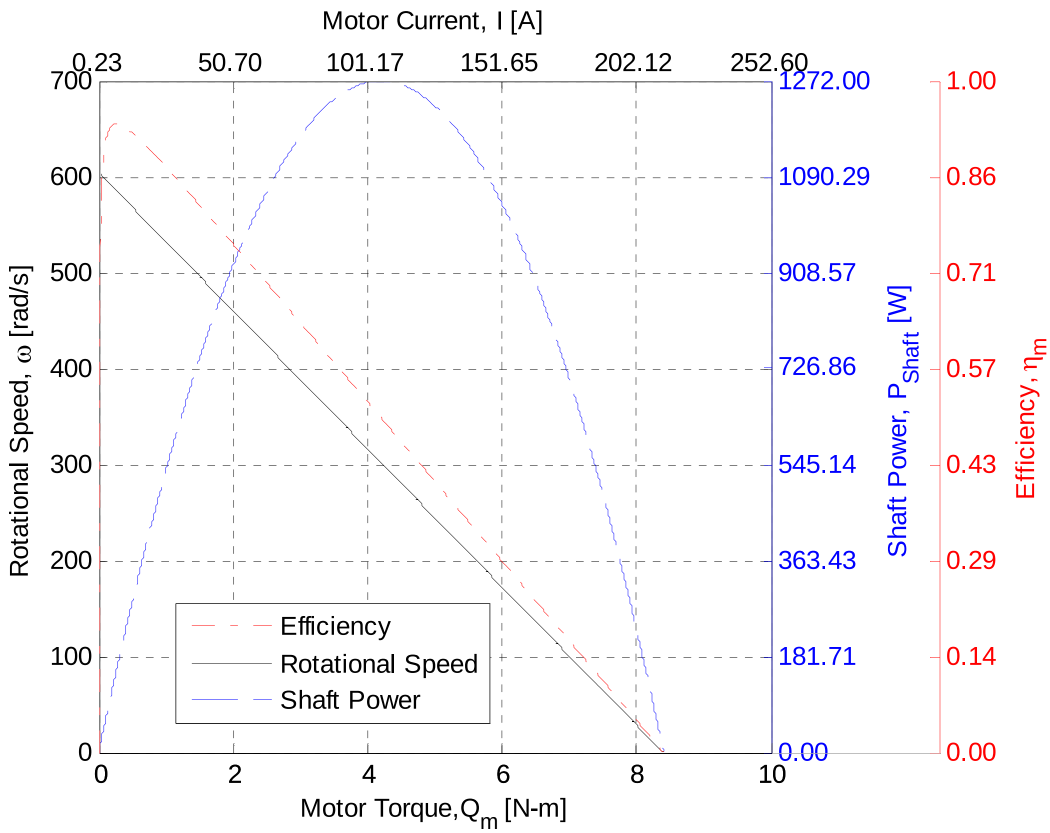


Figure 13: Electric Motor Map at Constant 24 Volts

$$I_{Stall} = \frac{Q_{Stall} - I_0}{K_v} = \frac{V}{R_m} \tag{14}$$

For a typical application, the EM will be operating between no-load and stall. Therefore, it is relevant to express the operating speed as a function of the no-load speed. Combining Equations 2 and 12 yields a relationship between the two.

$$\omega_m = \omega_0 - I_m R_m Kv = \omega_0 - (Q_m K_v^2 + I_0 K_v) R_m \tag{15}$$

The maximum output power is produced when the EM is loaded until the speed is reduced to half ω_0. The downside is that the EM will only be running at about 50 percent efficiency, so about half of the electrical power will be converted to heat. The EM will most likely overheat at this current and torque [31].

The EM efficiency is zero at no-load and at stall. Maximum power is also achieved when the supplied current is half I_{Stall}. The best efficiency always occurs when the supplied current is equal to the square root of the product of I_0 and I_{Stall}. The max EM efficiency (η_{Max}) can then be calculated [31].

$$I_{\eta_{m\_Max}} = \sqrt{I_0 * I_{Stall}} \tag{16}$$

$$\eta_{m\_Max} = \frac{I_{\eta_{m\_Max}} - I_0}{I_{\eta_{m\_Max}}} \tag{17}$$

The EM rotational speed for maximum efficiency is obtained by substituting Equations 16 into Equation 15.

$$\omega_{\eta_{m\_Max}} = \omega_0 - I_{\eta_{m\_Max}} R_m * Kv \tag{18}$$

Figure 13 was a good illustration of how the above concepts and equations apply, but an EM will only be operating at such high current and torque at start. After which, the current will be limited to a more suitable operating range. Figure 14 shows the same information as Figure 13 with the current operating range limited to 10 Amps. For this current range the output power does continue to increase with current as previously thought.

The previous plots depicted how an EM behaves at a constant input voltage with varying amounts of input current. It is also interesting to note how they behave at different input voltages. All the fore mentioned equations still apply; only the voltage will be allowed to vary. As shown in Figure 15, for a constant rotation speed, if the

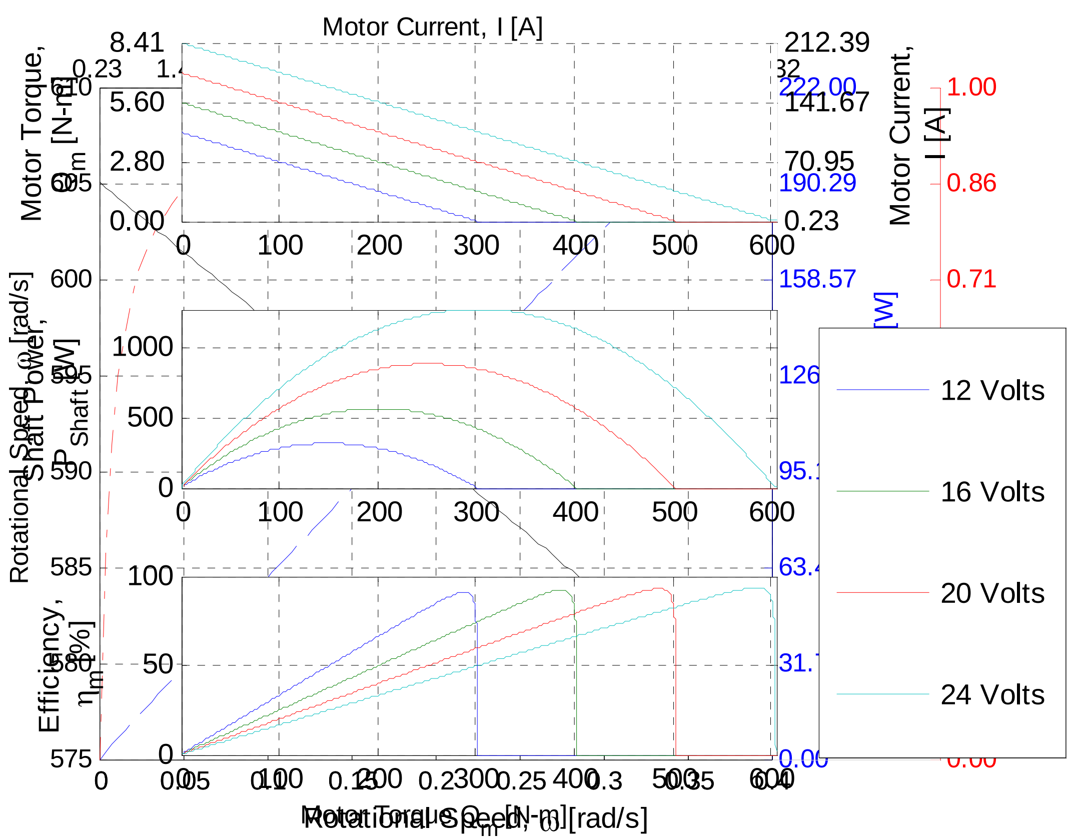


Figure 14: Electric Motor Map at Constant 24 Volts and Limited Current

voltage is decreased, the current and corresponding torque will decrease. Naturally, if the torque decreases with constant speed, then the shaft power will decrease. Also, since both the current and voltage decrease then the input power also decreases. It is, however, interesting that although both input and shaft power decrease, efficiency increases.

If constant power is required, then a decrease in voltage will cause an increase in current and torque. If the torque is increased, then the rotational speed has to decrease. It may be unclear as to why the efficiency decreases. From Equation 9, input power is equal to the shaft power plus the inefficiency terms. The first inefficiency increases as the square of the current. Since the current increased, the inefficiency increased further. The second inefficiency increases with rotational speed, but the rotational speed

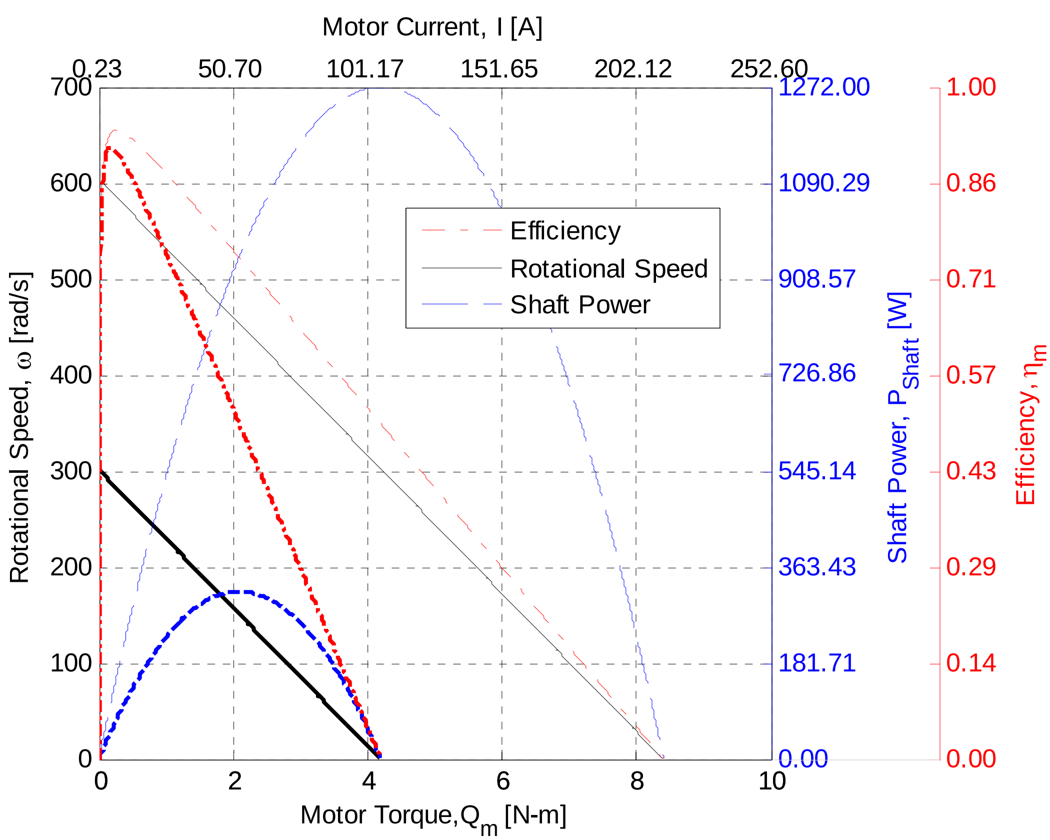


Figure 16: Electric Motor Map at Constant 12 and 24 Volts

decreased and therefore the second inefficiency decreased. Therefore, the input power increased. If the input power increases and the output power remains the same, then the efficiency decreases.

If constant torque is required, then the current will also be constant. A decrease in voltage with constant current leads to a decrease in input power and rotational speed and therefore, a decrease in shaft power. Since the output power has decreased more than the input power, the efficiency also decreases. Another way to look at the constant torque or current case is to re-plot Figure 13 and Figure 14 with two voltages. Figure 16 shows rotational speed, shaft power, and efficiency as function of current or torque for a constant 12 and 24 volts. The thicker lines in the lower left hand corner are in reference

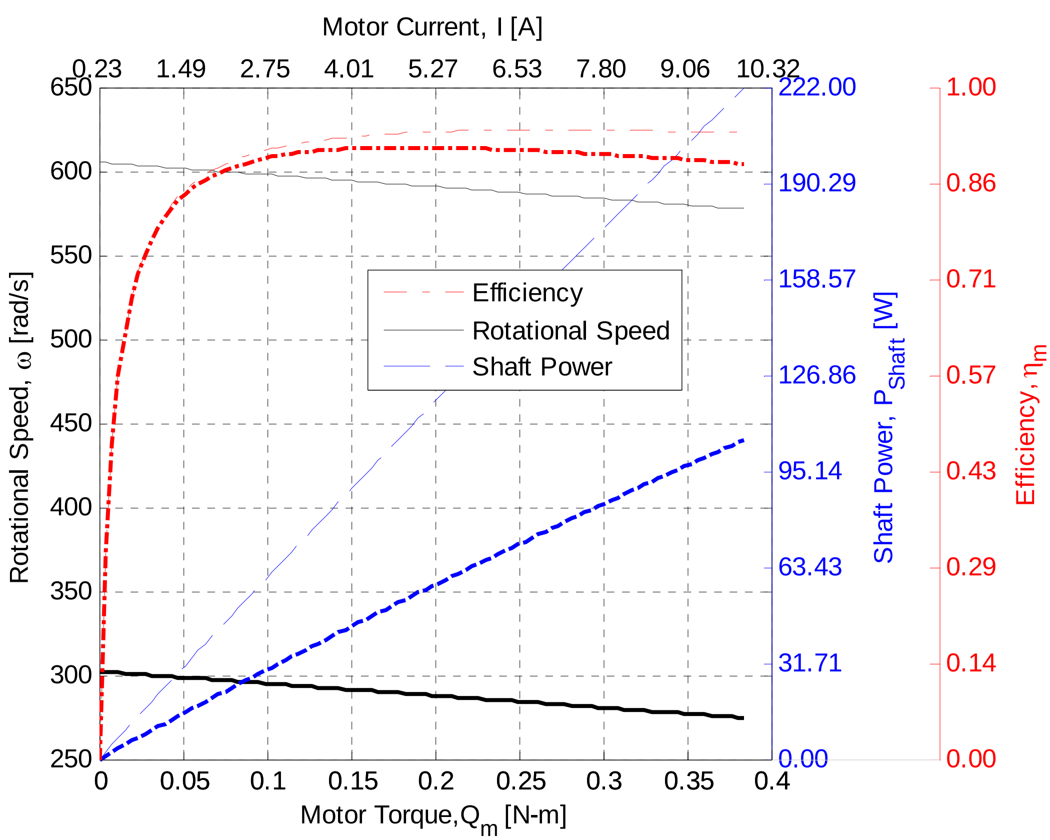


Figure 17: Electric Motor Map at Constant 12 and 24 Volts with Limited Current

to 12 volts and the thinner lines are in reference to 24 volts. As shown, at a constant current, the 24 volt system has a greater rotational speed, higher output shaft power and higher efficiency over the shared current range. Once again, Figure 16 is representative of the entire current range up to the stall current. Figure 17 shows the same 12 and 24 volt systems at a much more representative current range. As shown in Figure 17, at the low current, there is little difference in the efficiency of the two systems.

3.2. Propellers

A propeller is much like a wing in the fact that they are both made up of airfoil sections used to generate an aerodynamic force. A propeller is essentially a wing oriented perpendicular to the aircraft's longitudinal axis that rotates to produce thrust instead of lift. Also like a wing, propellers create drag and therefore have inefficiencies. For this reason, power sources of the propeller must be able to produce more power than is required by the propeller for the given flight condition. The power available to propel the vehicle, or the power produced by the propeller (P_p), is equal to the product of the power supplied by the shaft (P_{Shaft}) and the efficiency of the propeller (η_p) as well as the product of the forward velocity (V_∞) and the thrust produced by the propeller (T_p) [32][33].

$$P_p = \eta_p P_{Shaft} = T_p V_\infty \tag{19}$$

$$\eta_p = \frac{P_p}{P_{Shaft}} = \frac{T_p V_\infty}{P_{Shaft}} \tag{20}$$

Propeller efficiency is not constant. It varies with V_∞ and rotational speed n. As the propeller rotates through one circle the vehicle advances a distance (V_∞/n). The advance ratio (J) is then the ratio of that advance distance to the propeller's diameter (D).

$$J = \frac{V_\infty}{nD} \tag{21}$$

Figure 18 shows how the η_p of an APC 16-12E propeller changes with J. At low J, η_p is low and peaks at around 80 percent at J of about 0.7. Other standard propulsion equations have been developed to calculate non-dimensional performance parameters to compare experimental data of propellers of similar size and shape. These parameters include the coefficient of thrust (C_T), coefficient of power (C_P), and the coefficient of torque (C_Q) which are all functions of the air density (ρ_∞), n, and D. η_p can be expressed as a function of J, C_T, and C_p [34].

$$C_T = \frac{T_p}{\rho_\infty n^2 D^4} \tag{22}$$

$$C_P = \frac{P_{Shaft}}{\rho_\infty n^3 D^5} \tag{23}$$

$$C_Q = \frac{Q_p}{\rho_\infty n^2 D^5} = \frac{C_P}{2\pi} \tag{24}$$

$$\eta_p = J\frac{C_T}{C_P} \tag{25}$$

In recent years, there have been numerous studies to characterize the performance of low Reynolds number propellers for use on radio control (R/C) aircraft and RPA [35][36][37][38]. An example of such experimental data gathered at Wichita State University is shown in Figure 18.

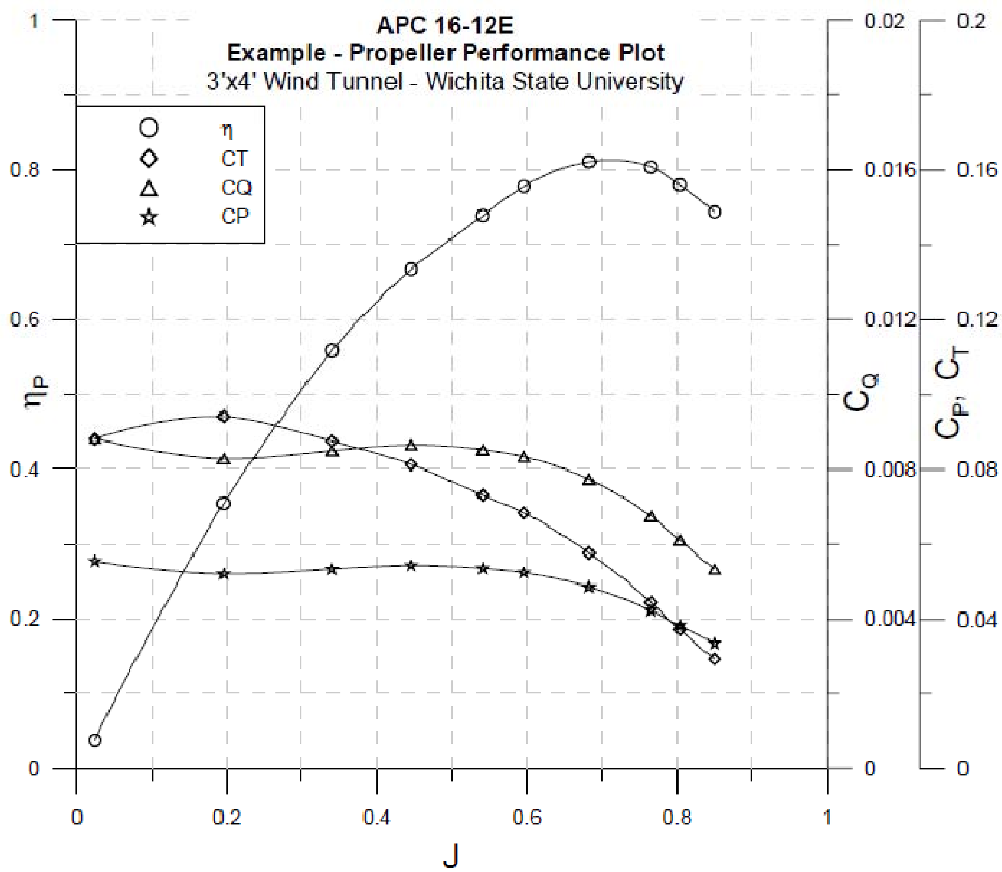


Figure 18: Experimental Performance plots for APC 16-12E propeller [35]

For an optimal propeller driven aircraft, the propeller and its power source need to operate at their peak efficiencies. If there is a mismatch in the optimal rotational speed between the two, then gearing can be used to better align the shaft's rotational speed with that due to the torque demand of the propeller. While gearing will align the speeds, it will also change the torque load on the power source. If the speed is increased then the torque decreases and vice versa. If the power source is an EM, then an increase in torque relates to an increase in current and an increase in system heat which may exceed the EM operating range. Gearing also adds more weight, more cost, its own inefficiencies, and possibly less reliability, so a different motor-propeller combination may be the better solution.

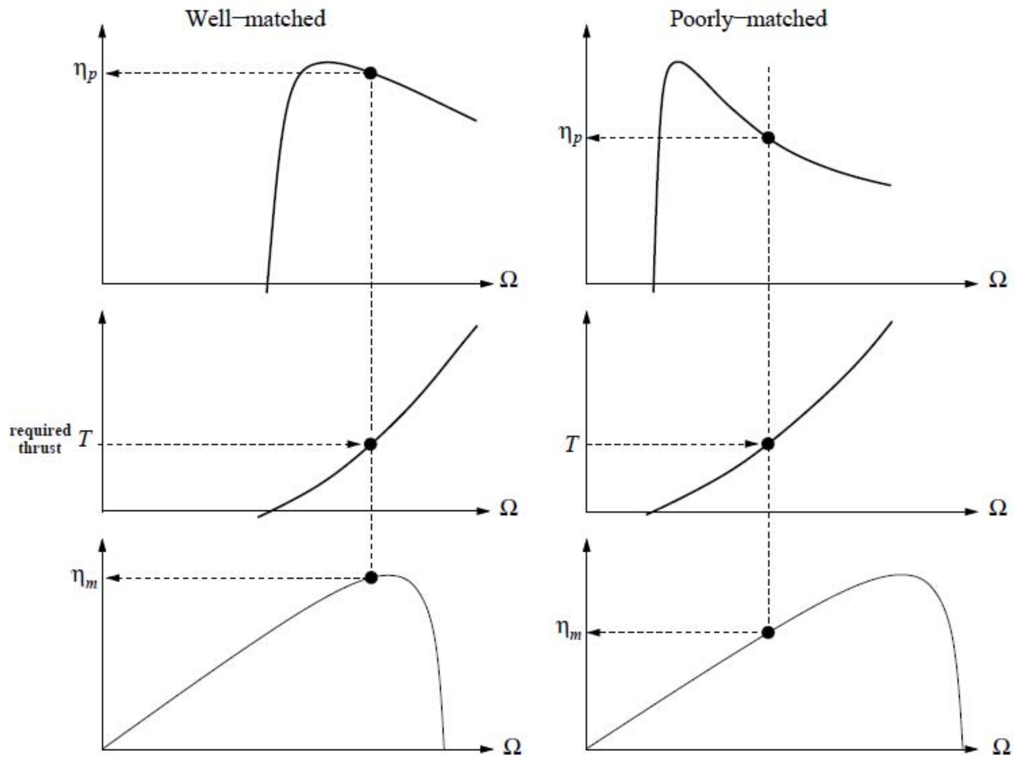


Figure 19: Well-Matched and Poorly-Matched Motor and Propeller Pairs

The tradeoff between a well-matched and a poorly-matched motor-propeller pair is demonstrated in Figure 19 [27]. If the system is well-matched, then each component is operating at or near its peak efficiency. If the system is poorly-matched, one or even both components are operating at much lower efficiencies and therefore, power is wasted to provide the same required thrust. In Figure 19, Drela denotes rotational speed as Ω. For a RPA designed for max endurance, in an EM only mode, properly matching the EM and propeller is the most important aspect of the H-EPS.

3.3. Internal Combustion Engines

An ICE uses the explosive combustion of fuel with an oxidizer to push a piston within a cylinder. The linear motion of the piston in the cylinder is converted to rotational motion through the connecting rod and crankshaft. Most reciprocating engines

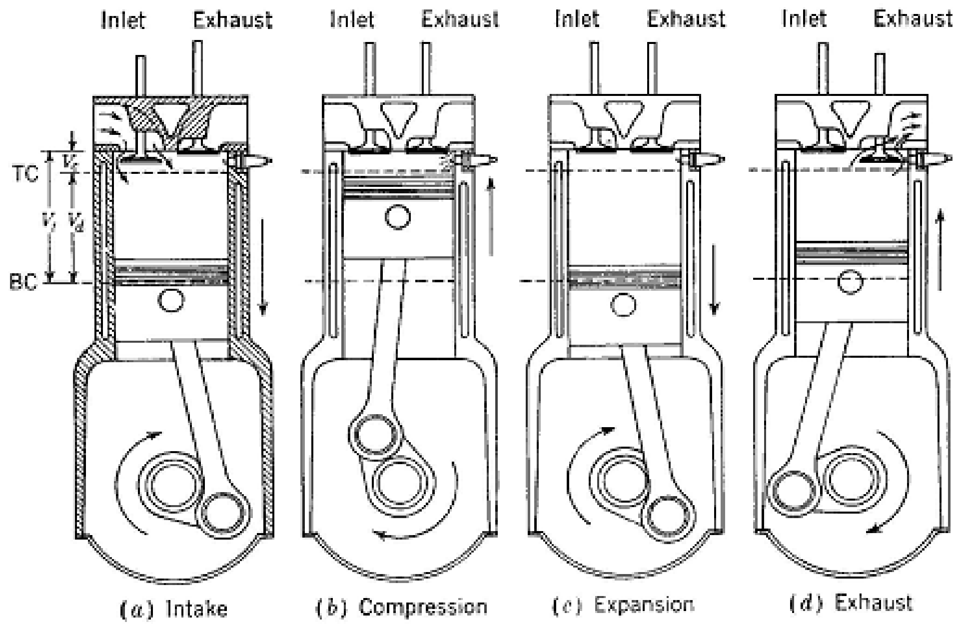


Figure 20: Four-Stroke Operating Cycle [39]

operate on a four-stroke cycle. One cycle consists of four strokes of a piston and two rotations of the crankshaft. As shown in Figure 20, the four-stroke engine consists of an intake stroke, a compression stroke, a power stroke and an exhaust stroke [39]. In order to obtain a higher power-to-weight ratio, the two-stroke engine was developed. Instead of using separate inlet and exhaust valves, the piston acts as a single valve letting air in and exhaust out and therefore, the two-cycle engine only needs one crankshaft revolution for each power stroke. This process is depicted in Figure 21.

The effectiveness of an ICE to convert chemical energy to mechanical energy is known as the fuel conversion efficiency (η_f). The fuel conversion efficiency of the engine is a function of the amount of thermal energy released by the fuel during combustion, known as the heating value (Q_{HV}), and the efficiency of an engine to use fuel to produce work, known as the specific fuel consumption (SFC). SFC is a measure of the fuel flow rate ($\dot{m}_f$) per power output (P) [39].

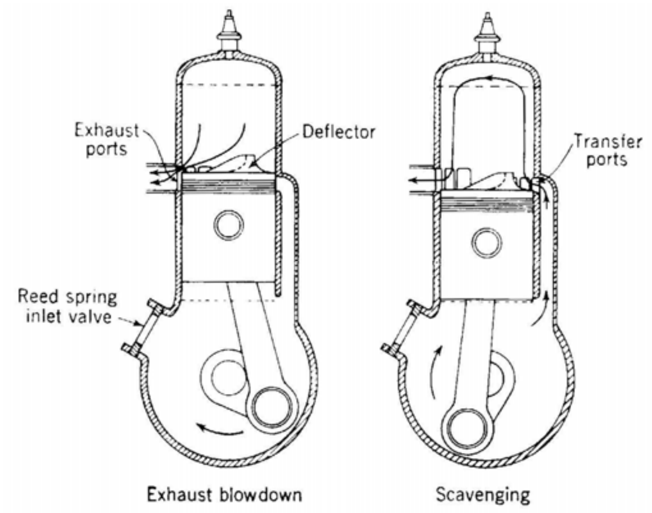


Figure 21: Two-Stroke Operating Cycle [39]

$$\eta_f = \frac{1}{SFC \cdot Q_{HV}} \tag{26}$$

$$SFC = \frac{\dot{m}_f}{P} \tag{27}$$

Commonly an engine's performance is described in terms of P and torque (Q) but a more useful relative engine performance measure is mean effective pressure (MEP). An engine's MEP is its work per cycle divided by the volume displaced (V_d) per cycle. This is the product of P and the crankshaft revolutions per power stroke (n_R) divided by the product of V_d and the crankshaft rotational speed (n). [39].

$$MEP = \frac{P n_R}{V_d n} \tag{28}$$

For four-stroke cycle engines, MEP can be expressed as

$$MEP = \eta_f \eta_v Q_{HV} \rho_\infty (F/A) \tag{29}$$

where, F/A is the fuel-to-air ratio. For two-stroke engines, MEP can be expressed as

$$MEP = \eta_f \eta_{tr}\, \Lambda Q_{HV} \rho_\infty (F/A) \tag{30}$$

where, $\eta_{tr}$is the trapping efficiency and Λ is the delivery ratio.

$$\eta_{tr} = \frac{mass\ of\ delivered\ air\ (or\ mixture) retained}{mass\ of\ delivered\ air\ (or\ mixture)} \tag{31}$$

$$\Lambda = \frac{\text{mass of delivered air (or mixture) per cycle}}{\text{displaced volume} * \text{ambient air (or mixture)density}} \tag{32}$$

An engine's power and torque can be expressed in terms of MEP.

$$P = \frac{MEP \cdot n \cdot V_d}{n_R} \tag{33}$$

$$Q = \frac{MEP \cdot V_d}{2\pi n_R} \tag{34}$$

Figure 22 shows a typical ICE power map as a function of rotational speed (N) and Q. Figure 24 shows a typical ICE fuel consumption map as function of N and Q. Figure 23 shows a typical ICE efficiency map as function of N and Q. If the N and Q of an ICE are known then the efficiency of the ICE can be determined.

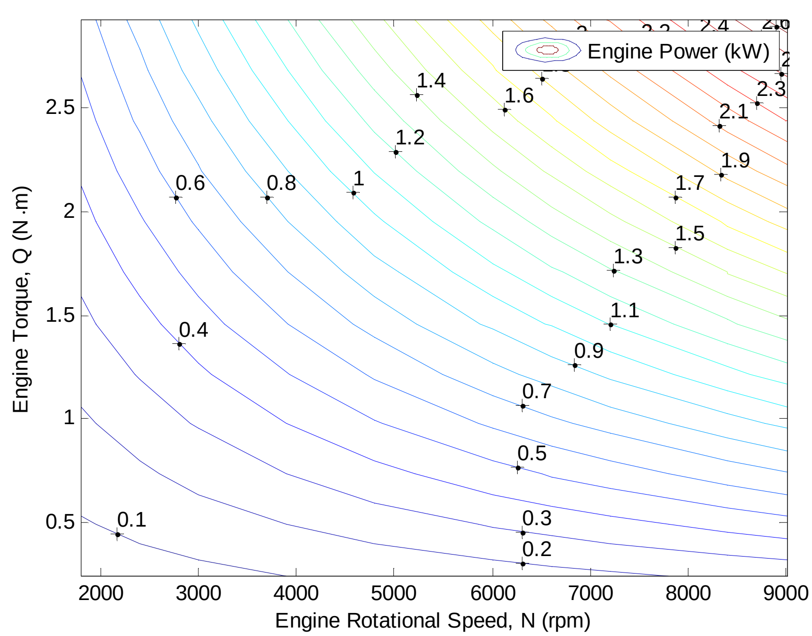


Figure 22: Engine Power Map

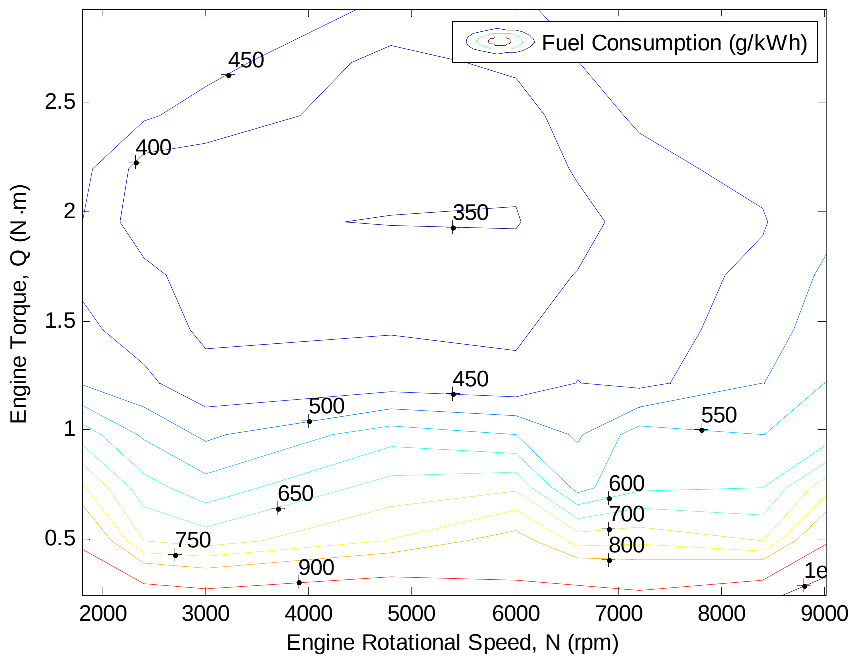


Figure 24: Engine Fuel Consumption Map

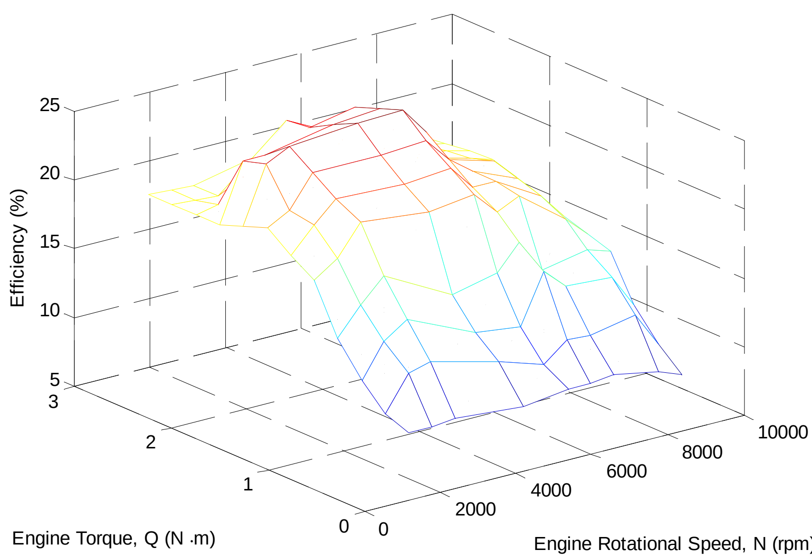


Figure 23: Engine Efficiency Map

III. Methodology

1. Chapter Overview

This chapter outlines the methodology used by the author to formulate an analysis tool, using constrained static optimization, to size the H-EPS components. Hiserote outlined three different parallel hybrid-electric configurations: clutch-start parallel hybrid-electric configuration (Figure 25), electric-start parallel hybrid configuration (Figure 26), and centerline-thrust hybrid configuration (Figure 27). If it is assumed that the ICE is allowed to idle when not in use, then the clutch-start and electric-start configurations can be treated as the same; the difference being the efficiency of the clutch/one-way bearing when the ICE is powering the system. The analysis tool optimizes this type of configuration due to the fact that it only incorporates one propeller as opposed to the centerline-thrust configuration.

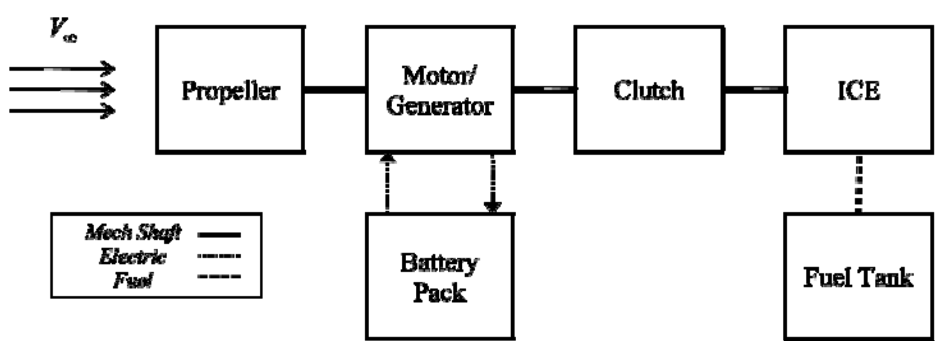


Figure 25: Clutch-Start Parallel Hybrid-Electric Configuration [4]

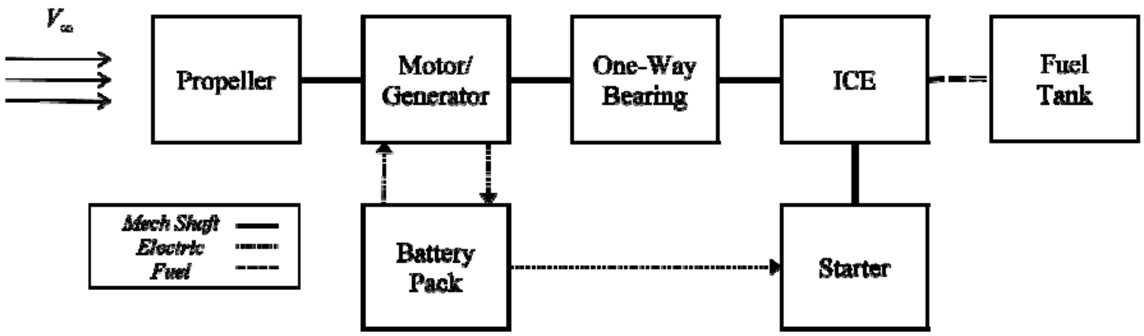


Figure 26: Electric-Start Parallel Hybrid-Electric Configuration [4]

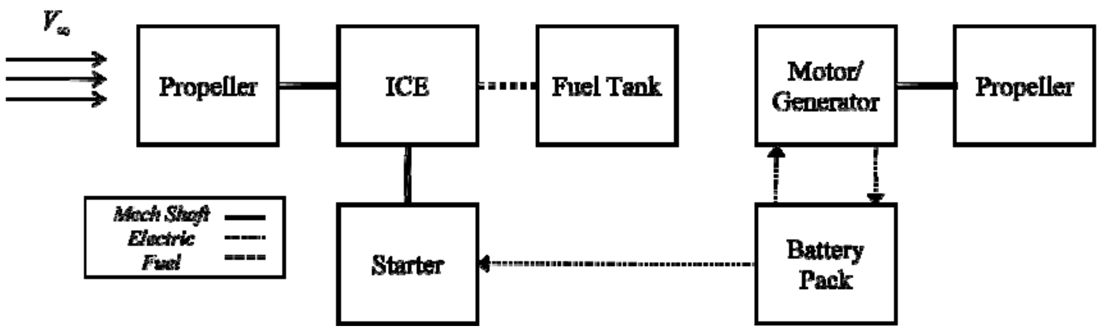


Figure 27: Centerline-Thrust Hybrid Configuration [4]

2. System Optimization Process

The driving force behind an aircraft propulsion system is the thrust required to propel the vehicle through the air. The output torque and rotational speed the propulsion system must provide is driven by the torque and rotational speed required of the propeller to produce the required thrust. Unlike a conventional parallel H-EPS that is capable of using both power sources to produce the required torque, the RPA mission dictates that the EM alone must be capable of producing the required torque for endurance. For this reason, the goal of the optimization process, outlined in Figure 28, is to maximize the operating efficiency of an EM and propeller combination.

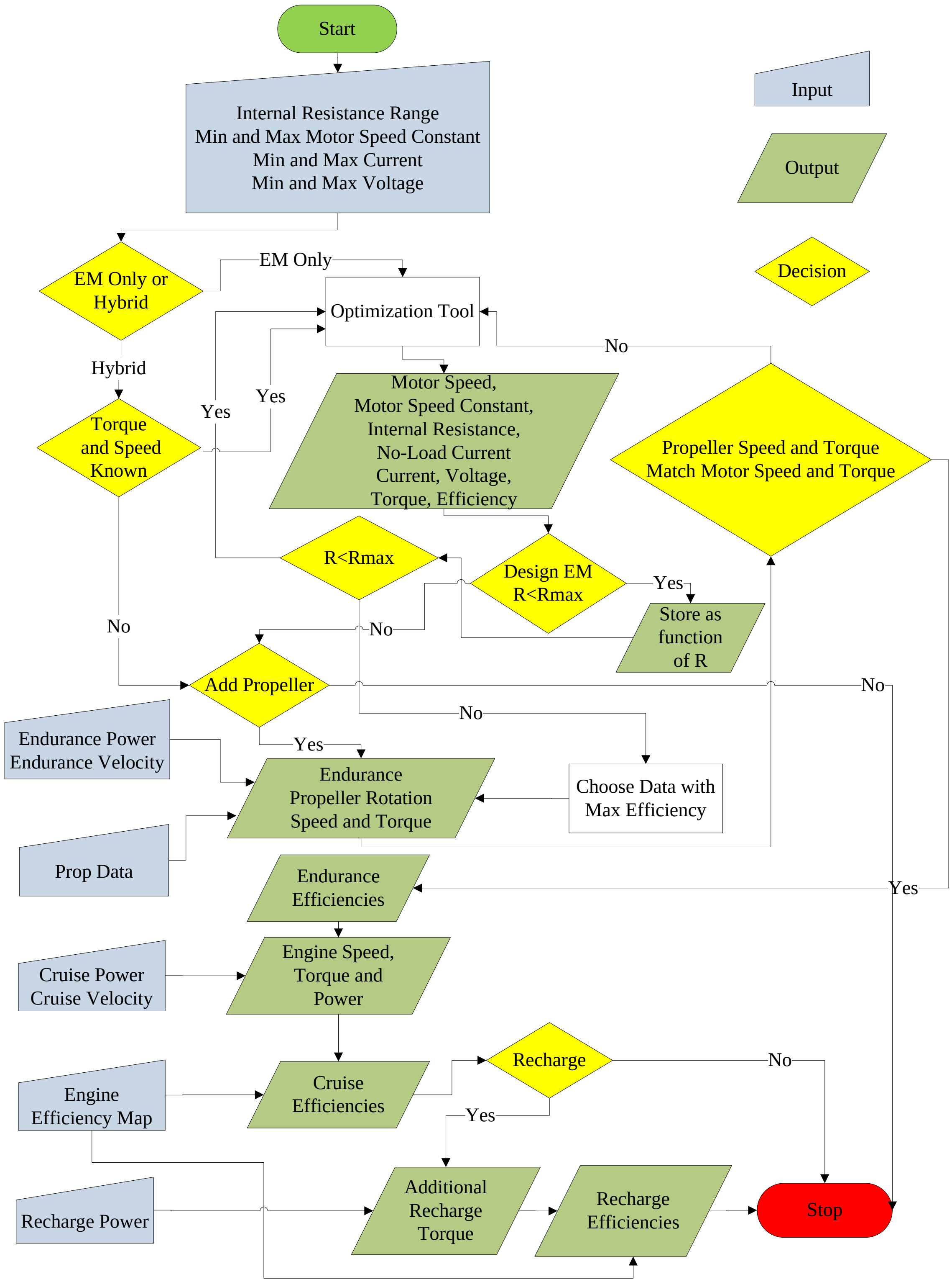


Figure 28: Parallel Hybrid-Electric Propulsion Component Optimization Flowchart

In general, optimization can be defined as minimizing or maximizing an objective function subject to constraints on the design variables [40]. For the proposed H-EPS, the most critical mission segment is the all-electric endurance. In order to maximize the endurance of any RPA, the EM efficiency must be maximized. Operating at maximum efficiency will require less current from the batteries and therefore, result in increased endurance. The cost function (CF) for the H-EPS component optimization is the EM efficiency at the endurance speed. Substituting Equation 2 into Equation 11 yields efficiency as a function of rotational speed rather than voltage.

$$CF = \eta_m = \frac{(I - I_o)\frac{\omega_m}{K_v}}{\left(\frac{\omega_m}{K_v} + IR_m\right)I} \tag{35}$$

Equation 35 yields the cost function as a function of five design variables. If the no-load current and the internal resistance can be treated as constants, then the cost function contains only three design variables. Ideally, they cannot be treated as constants. EMs do not all have the same no-load current and resistance. Gur and Rosen examined over 1500 brushed and brushless DC motors and examined the relationship between no-load current and internal resistance. Their results are shown in Figure 29. They used the following relationship to relate their findings.

$$I_0 = \frac{B_{I0}}{(R_a)^{0.6}} \tag{36}$$

For the purpose of this thesis, the lower line approximation was used. There still has to be an assumption of the internal resistance, and although not exact, it is a good place to start.

$$I_o = \frac{0.1}{R_m^{0.6}} \tag{37}$$

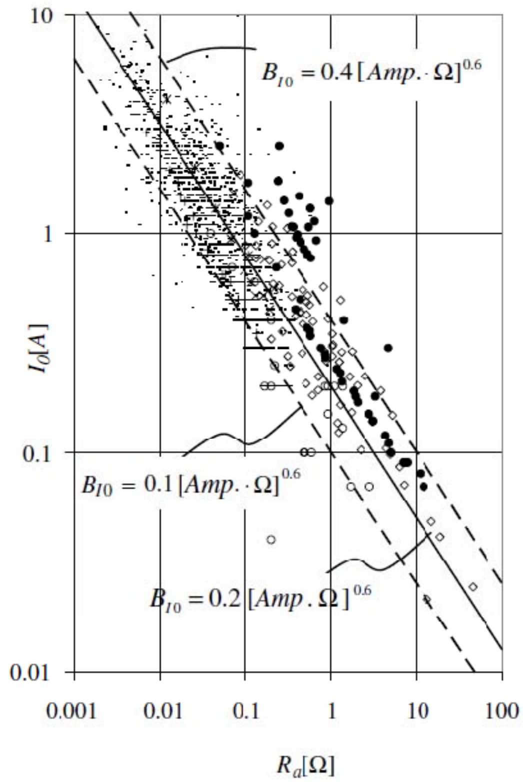


Figure 29: No-Load Current as a Function of Internal Resistance [41]

The cost function must be constrained to ensure that the optimization process converges on a minimum value [40]. Since the desired output is maximum efficiency, the cost function will be treated as negative and therefore, the optimization process will converge on the most negative efficiency. Current is constrained between the maximum system's current (or the manufacturer specified nominal current) and the no-load current. Voltage is constrained between the maximum system voltage (or the manufacturer specified nominal voltage) and the minimum voltage necessary to supply the required power. The motor speed constant is constrained based on an average low and high motor speed constant for a wide range of EMs. Although torque is not a design variable, it can also constrains the system since current is a function of torque, as shown in Equation 5. Once a propeller is added to the EM, the torque required by the propeller will constrain the current. When testing the EM against the manufacturer's specification, the minimum and maximum torque can be set to an extreme low and high value. EM rotation speed may be constrained by operating limits or by the propeller as well.

Table 1: Cost Function Constraints

Minimum Value	Parameter		Maximum Value
I_0	Current	I	I_{max}
$\frac{\omega_{min}}{K_v} + I_{min}R_m$	Voltage	V	$\frac{\omega_{max}}{K_v} + I_{max}R_m$
$K_{v_{min}}$	Motor Speed Constant	K_v	$K_{v_{max}}$
$\frac{I_{min} - I_0}{K_v}$	Motor Torque	Q_m	$\frac{I_{max} - I_0}{K_v}$
Propeller Torque and *GR*			Propeller Torque and *GR*
$(V_{min} - I_{max}R_m)K_v$	Motor Rotation Speed	ω_m	$(V_{max} - I_{min}R_m)K_v$
Propeller Speed and *GR*			Propeller Speed and *GR*
GR_{min}	Gear Ratio	GR	GR_{max}

One option of the analysis tool allows for optimizing a gear ratio. For that option, gear ratio does not become another design variable but rather increases the EM speed and decreases the EM torque required to rotate the propeller. Essentially, the gear ratio is reducing the current requirement by allowing the voltage to approach the system or defined maximum. Maximum gear ratio may be constrained by available space. The analysis tool only takes into account user input gear ratio limits. Table 1 outlines the design variable constraints and how they are determined.

3. Analysis Tool Inputs

There are several inputs the user is required to make prior to starting the analysis process. The inputs are self-explanatory and vary based on which case the user chooses. Figure 30-Figure 38 are snapshots, or the different sections where the user is required to make inputs.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Batteries
V_bat=25.9;%        [V]         Single Battery Pack Voltage
Num_bat=9;%         [#]         Number of Battery Packs
C_bat=3300;%        [mA/r]      Battery Capacity
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 30: Input Battery Data

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%DC-DC Convertor Output/System Maximum Voltage and Current
Imax=30;%         [A]         Max System Current
Vmax=40;%         [V]         Max System Voltage
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 31: Input System Maximum Current and Voltage

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Gear Ratio
%%%Optimized Gear Ratio
if GR_Case==1%
    GRmax=10;%      [#] Max EM/Prop Shaft Gear Ratio
    GRmin=0.25;%    [#] Min EM/Prop Shaft Gear Ratio

%%% Input Gear Ratio
elseif GR_Case==2
    EM_Gear_Teeth=22;%    [#]   Number of teeth on EM Gear
    Shaft_Gear_Teeth=32;%[#]   Number of teeth on Shaft Gear
    GR=Shaft_Gear_Teeth/EM_Gear_Teeth;%[#/1]   EM/Prop Shaft Gear Ratio
    GR=1;%                [#]   User input gear ratio or GR
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 32: Input Gear Ratio

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Electric Motor for Endurance
Pend=124.2;%        [W]         Power Required for cruise
Vend=28;%           [knots]     Endurance Velocity 25-30 knots
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 33: Input Endurance Requirements

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Electric motor internal resistance range for designing the optiminal EM
R_Range=(.01:.01:2);%     [ohms]
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 34: Input Resistance Range for Designing Electric Motor

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Know EM Specifications
if EM_Case==1
    R=0.113;%            [ohms]      Electric Motor Internal Resistance
    I0=0.225;%           [Amps]      Known Electric Motor No-Load Current
    EM_NC=9.15;%         [V]         Known Electric Motor Nominal Curnet
    EM_NV=24;%           [V]         Known Electric Motor Nominal Voltage
    Kvmax_N=241;%        [rpm/V]     Electric motor speed constant
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 35: Input Known Electric Motor Specifications

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Internal Combustion Engine and Cruise
Pcruise=265.7;%      [W]         Power Required for cruise
Vcruise=40;%         [knots]     Cruise Velocity 40-50 knots
Nmax_ICE=16000;%     [rpm]       Max internal combustion engine rpm
Nmin_ICE=4000;%      [rpm]       Min internal combustion engine rpm
N_ICE=5000;%         [rpm]       Optimum internal combustion engine Speed
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 36: Input Cruise Requirements and Internal Combustion Engine Data

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Clutch Efficiency
Eff_Clutch=.99;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 37: Input Clutch Efficiency

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Regeneration Power Required
PGen-75;%        [W]         Power Required for generator
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 38: Input Required Regeneration Power

4. Analysis Tool Options

The flowchart in Figure 28 is a decision tree based on what the user intends to use the analysis tool for. The analysis tool options are shown in Figure 39. The first option, Test_Case, allows the user to select whether they want to test an EM only or test an EM

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%Analysis Tool Options%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Test_Case=2;%      1=Electric Motor Only;       2=Hybrid Configuration
EM_Case=2;%        1=Test Electric Motor        2=Design Electric Motor;
GR_Case=2;%        1=Optimized Gear Ratio       2=Input Gear Ratio
Plot_Switch=1;%    1=Plots;                     2=No Plots
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Figure 39: Analysis Tool Options

and propeller together as part of a hybrid system. The second option, EM_Case, allows the user to either test the system with both a COTS EM and propeller or design an EM, for maximum efficiency, based on performance requirements and a COTS propeller. The third option, GR_Case, allows the user to either determine the optimal gear ratio or input a predetermined gear ratio. This analysis tool creates a great number of plots and figures; most of which just show graphically what the tool is doing mathematically. The tool will run with or without creating the plots. The fourth and final option, Plot_Switch, allows the user to turn off the plots in exchange for allowing the tool to run faster. Each option is discussed in more detail in the following sections.

4.1. Test_Case Option

The Test_Case option allows the user to select whether they want to test an EM only or test an EM and propeller together as part of a hybrid system. The main purpose of the analysis tool is to optimize the hybrid configuration, but it may be best to first take a step back and start with examining the COTS EM. The EM only flowchart is traced in blue in Figure 40.

Some manufacturers supply the motor specification for their specific motors, while others may only supply part of the data, if at all. Table 2 lists the motor specification data for four Maxon motors. If the manufacturer does not provide the data, then the no-load current (item 3), internal resistance (item 10), and motor speed constant

Table 2: Maxon Motor Specification Sheet [42]

			370354	370355	370356	370357
Motor Data (provisional)						
	Values at nominal voltage					
1	Nominal voltage	V	24	36	48	70
2	No load speed	rpm	5780	5780	4870	2810
3	No load current	mA	225	150	87.4	28
4	Nominal speed	rpm	5540	5540	4610	2540
5	Nominal torque (max. continuous torque)	mNm	354	354	362	385
6	Nominal current (max. continuous current)	A	9.15	6.10	3.93	1.65
7	Stall torque	mNm	8420	8420	6960	4090
8	Starting current	A	212	142	73.9	17.2
9	Max. efficiency	%	94	94	93	92
	Characteristics					
10	Terminal resistance	Ω	0.113	0.254	0.649	4.070
11	Terminal inductance	mH	0.0937	0.211	0.528	3.370
12	Torque constant	mNm / A	39.6	59.4	94.1	238
13	Speed constant	rpm / V	241	161	101	40.2
14	Speed / torque gradient	rpm / mNm	0.687	0.687	0.700	0.687
15	Mechanical time constant	ms	4.20	4.20	4.24	4.20
16	Rotor inertia	gcm^2	584	584	578	584

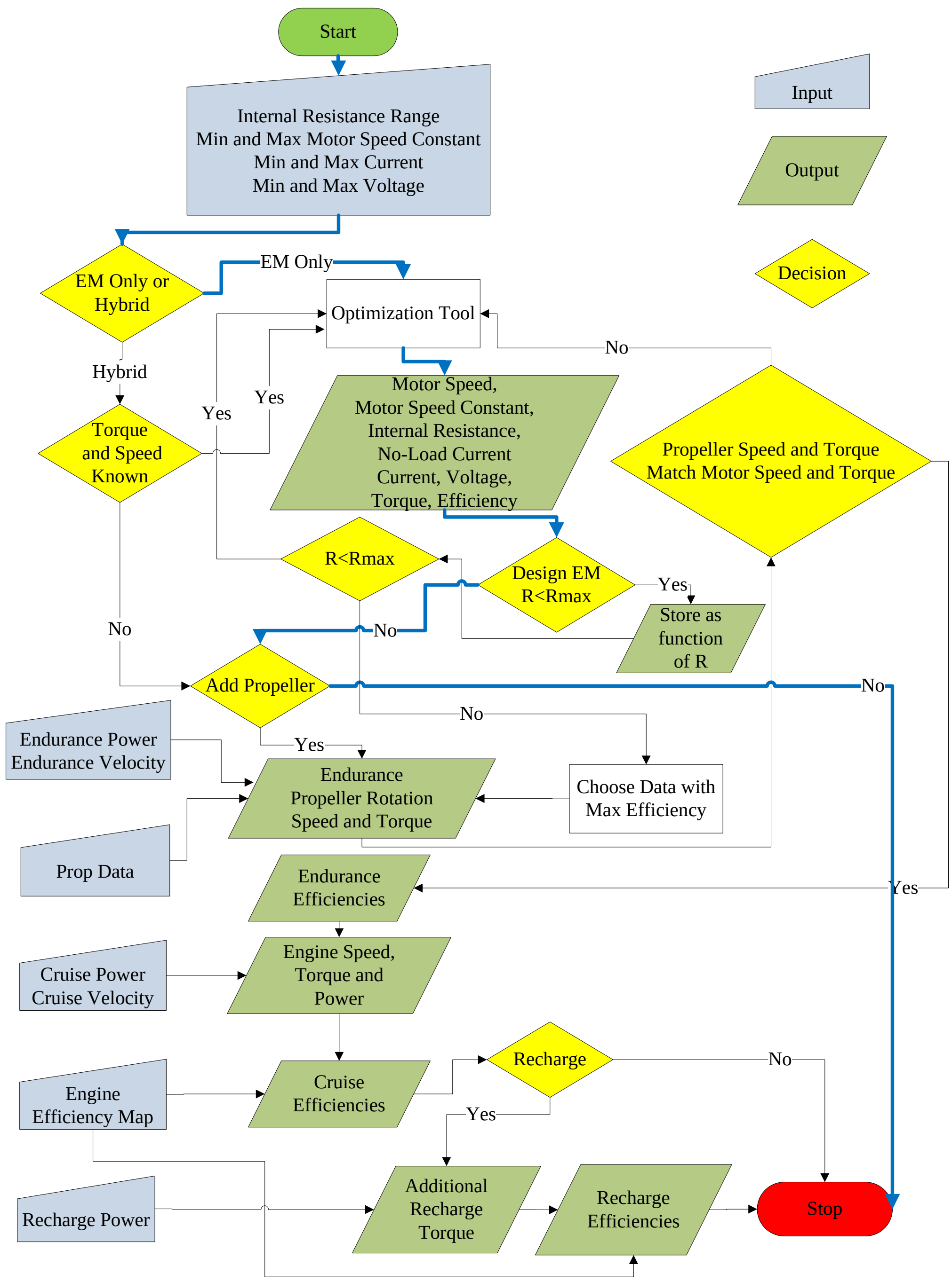


Figure 40: Optimization Flowchart with EM Only Mode in Blue

(item 13) can be determined experimentally. By combining those values with the nominal voltage (item 1) and nominal (max continuous) current (item 6), all the other values, except for items 11, 15, and 16 can be calculated based on equations in section II.3.1. The resultant data produces plots similar to those in section II.3.1. If the EM is well understood, then the user can select to test the hybrid configuration.

4.2. EM_Case Option

The EM_Case option allows the user to test the hybrid configuration with either a COTS EM or use the optimization process to design an EM for maximum efficiency

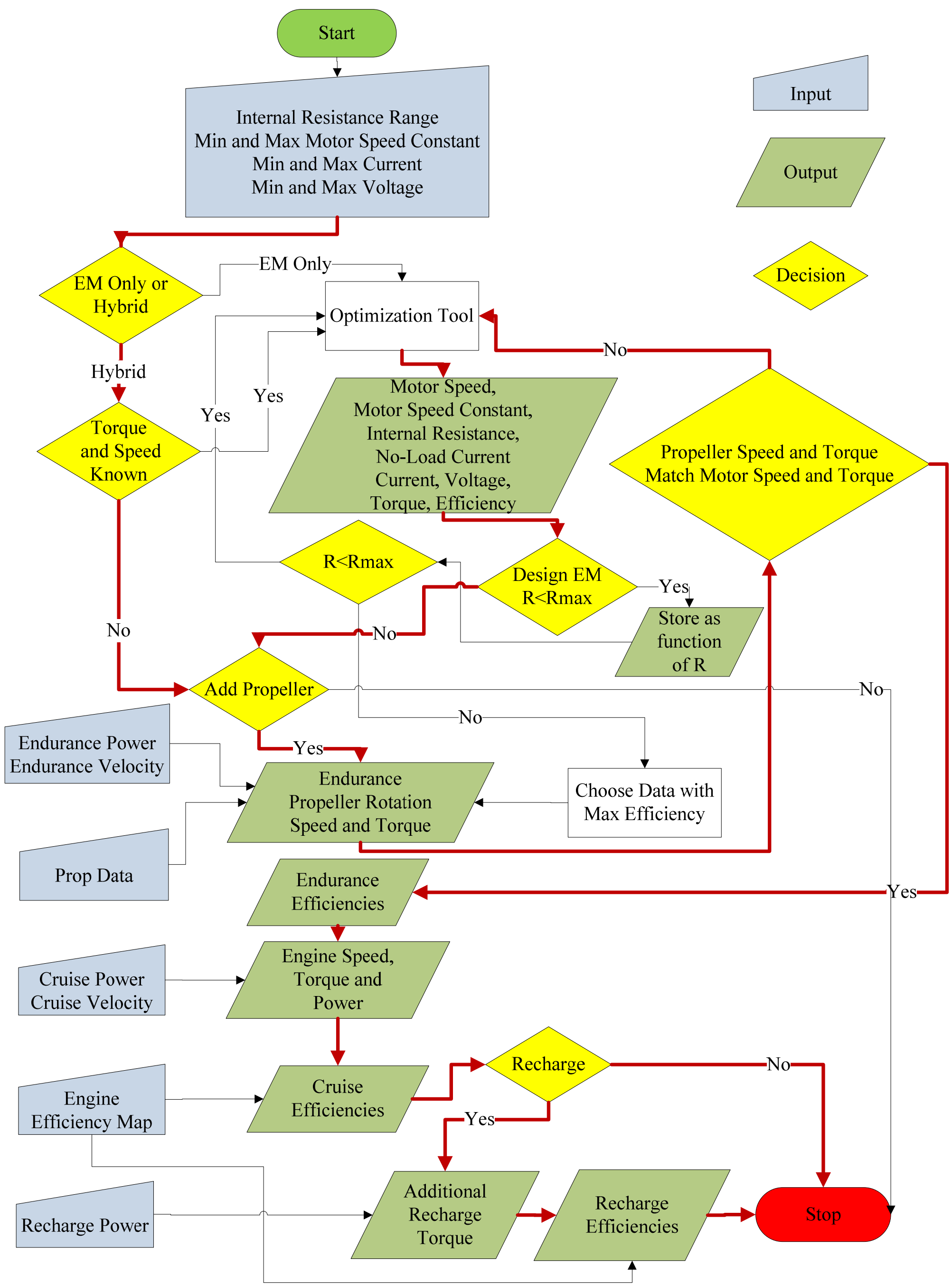


Figure 41: Optimization Flowchart with COTS EM Hybrid Mode in Red

based on performance requirements and a COTS propeller. Figure 41 shows the path the analysis tool used to match a COTS EM and propeller. When testing a COTS EM, the internal resistance, no-load current and motor speed constant are all known, therefore, their minimum and maximum values are exactly equal to the known value. The current and voltage range is set by the user-defined system. In order to calculate the optimized current and voltage, a propeller is added to define the required torque and rotational speed. The required torque, rotational speed, current, and voltage determine the EM operating efficiency and together with the propeller efficiency at endurance, the total endurance efficiency is determined. The rest of the analysis tool path is the same for all other cases and is discussed in the next section.

Another option of EM_Case allows the user to design an EM based on a given propeller. Figure 42 shows the path the analysis tool uses to design an EM for maximum efficiency based on the selected propeller. For this case, neither the no-load current, internal resistance nor the motor speed constant is known. To start, the internal resistance is defined as a range. The resistance range is based on market data and is set wide to allow the analysis tool to zero in on the most efficient EM. The no-load current is then estimated based on Figure 29 and Equation 37. If the required propeller torque and speed are known, then the analysis proceeds to the optimization tool. If not, then a propeller must be selected. The output of the optimization tool is the EM specifications and efficiency based on the particular internal resistance. These values are stored and then the internal resistance is increased by the predetermined step size. The process is repeated until the entire internal resistance range has been evaluated. Then, the designed EM that produced the highest efficiency is selected and processed through the optimization tool one final time so it can proceed through the rest of the flowchart. It is important to note that the designed EM does not require a gear ratio. By design, the gear ratio is one. This EM would be most suited for an inline configuration. The designed EM could be utilized in a non-inline configuration if the design dictates.

The propeller speed and torque required for cruise is determined based on the cruise performance requirements and the pre-selected propeller. Evaluating the ICE efficiency map, at the required cruise torque and rotational speed, reveals the efficiency of the ICE under the cruise conditions. The ICE efficiency together with the propeller efficiency at cruise determines the total cruise efficiency.

When the aircraft is cruising, with the ICE supplying power to the propeller, the EM can be used as a generator to recharge the batteries, by increasing the ICE output torque while maintaining the required rotational speed. Much like the EM converts current and voltage to torque and rotational speed; the generator converts torque and rotational speed to current and voltage. The specified generator output power (P_{Gen}) along with the cruise rotational speed will determine the generator output voltage and current. Just like in Equation 2 for the EM, the generator voltage (V_{Gen}) is determined by Equation 38. If the generator current (I_{Gen}) is expressed as P_{Gen} divided by I_{Gen} (Equation 39), then V_{Gen} is determined by Equation 40. Equation 40 yields two solutions; one positive and one negative. The real solution is the positive one.

$$V_{Gen} = \frac{\omega_{Gen}}{Kv} - I_{Gen}R_m \tag{38}$$

$$I_{Gen} = \frac{P_{Gen}}{V_{Gen}} \tag{39}$$

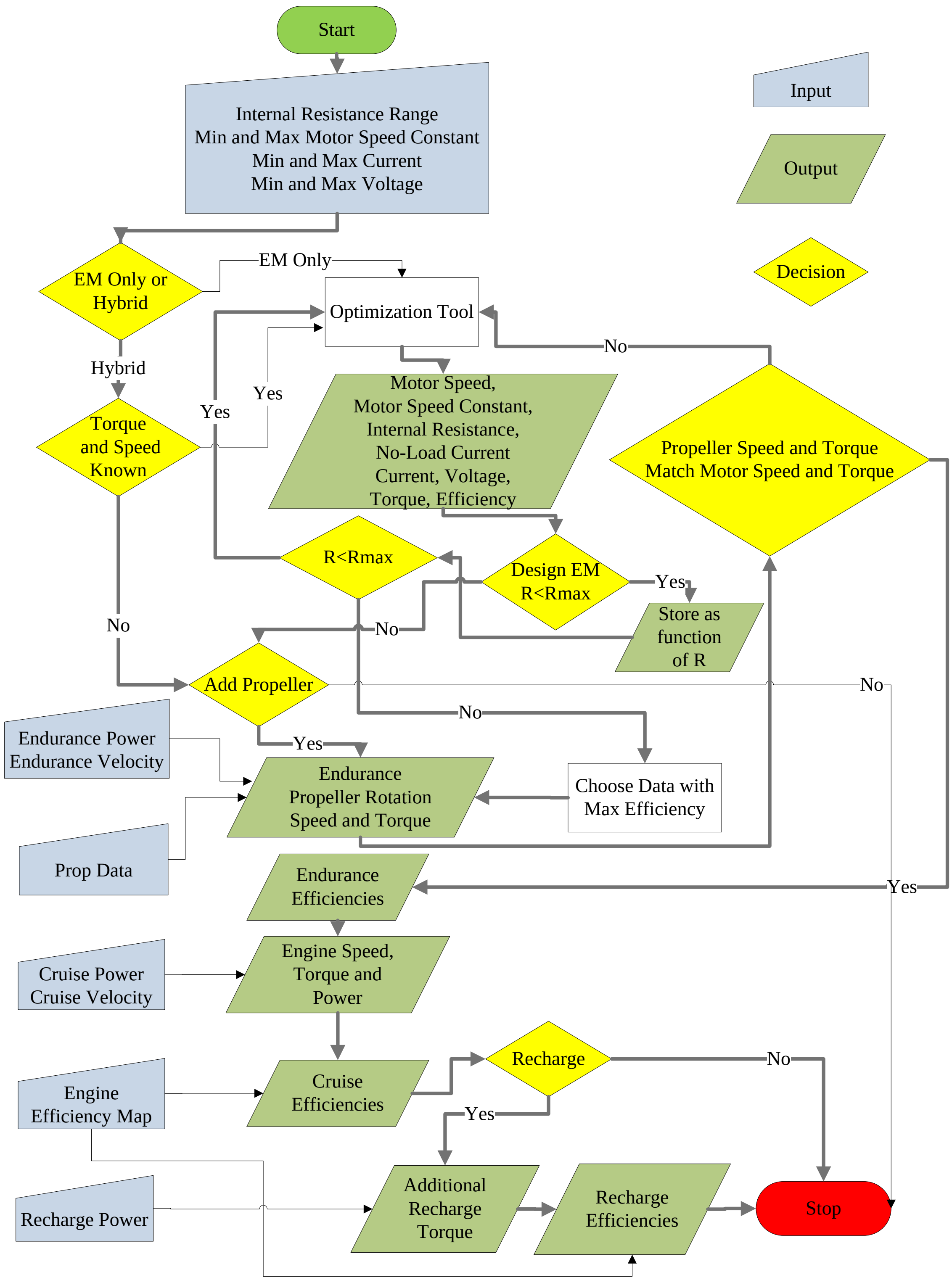


Figure 42: Component Optimization Flowchart with Designed EM Hybrid Mode in Grey

$$V_{Gen} = \frac{-\sqrt{\omega_{Gen}^2 - 4P_{Gen}K_v^2R_m} - \omega_{Gen}}{2K_v} \ or \ \frac{\sqrt{\omega_{Gen}^2 - 4P_{Gen}K_v^2R_m} + \omega_{Gen}}{2K_v} \tag{40}$$

With the generator voltage known, the generator current is determined by Equation 39. The generator current is the current required to charge the batteries at the specified voltage. The torque required by the generator to produce the required current is determined by Equation 41.

$$Q_{Gen} = \frac{I_{Gen} + I_0}{K_v} \tag{41}$$

The torque required by the generator is added to the torque required to turn the propeller and the sum is the output torque required from the ICE.

$$Q_{ICE} = Q_{prop} + Q_{Gen} \tag{42}$$

4.3. GR_Case Option

The GR_Case Option allows the user to choose between optimizing a gear ratio and manually inputting one. This case is used for testing a COTS EM and propeller combination as a designed EM does not require a gear ratio. If the user chooses to optimize the gear ratio, then the gear ratio bounds must be specified as listed in Table 1. If the user chooses to input a gear ratio, then it can be input in two ways. It can either be entered as a true ratio of two gears or simply as a decimal. If toothed gears are used, then they must be the same pitch and the gear ratio is input as the ratio of teeth. If smooth gears are used, then the gear ratio is the ratio of the diameters. The gear ratio input section is shown in Figure 32.

If the user inputs a gear ratio, and the required gear ratio is greater, then the analysis tool will display an error message with a suggested gear ratio. An example error message is shown in Figure 43. The suggested gear ratio is then entered as the user input gear ratio. If toothed gears are used, then the optimized gear ratio may not be possible and therefore, the closest gear ratio combination is used.

```
ERROR!
The torque required to rotate the propeller is
more than the EM can provide.
Either increase the max current, increase the
gear ratio to at least 1.1498, or try the
option to optimize the gear ratio.
Press Ctrl C to Exit
ERROR!
fx |
```

Figure 43: Gear Ratio Error

5.   Propeller Data

The analysis tool requires a separate MATLAB .m file of the propeller data in order to run. A sample APC 18x8 propeller file is shown in Figure 44. Any experimental propeller data file may be used, but it must be in the specific format.

```
function[Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APC18x8()
syms APC
Prop_Brand=APC;
        Prop_Dia=18;%            [in]        Propeller Diameter
        Prop_Pitch=8;%           [in/rev]    Propeller Pitch
        % Prop_Data=[J Eff CT CP]
        Prop_Data=...
            [0.0289 0.0869  0.0810  0.0269
            0.1036  0.2924  0.0809  0.0287
            0.2317  0.5523  0.0722  0.0303
            0.3786  0.7279  0.0569  0.0296
            0.3983  0.7385  0.0539  0.0291
            0.4706  0.7689  0.0447  0.0274
            0.5425  0.7523  0.0340  0.0245
            0.5921  0.7154  0.0265  0.0220
            0.6243  0.6673  0.0213  0.0200
            0.6510  0.6078  0.0169  0.0181];
end
```

Figure 44: APC 18x8 Propeller Data Function File

6. Experimental Setup

6.1. Dynamometer

The test setup utilized a small engine dynamometer built by the Land and Sea Corporation and controlled by their proprietary software [43]. The dynamometer uses a 96 V eddy-current magnetic-brake to apply a load to the rotating shaft. The applied torque was measured with a strain gage attached to cradle that holds the test setup. The strain gage and brake are shown in Figure 45. By using a strain gage, torque is measured independent of gear ratio. There is a 2:1 gear ratio between the dynamometer shaft and H-EPS shaft. Rotational speed is measured by counting the revolutions of the dynamometer gear and then is converted within the dynamometer software. The complete dynamometer setup is discussed in detail in Greiser [5] and Mengistu [44].

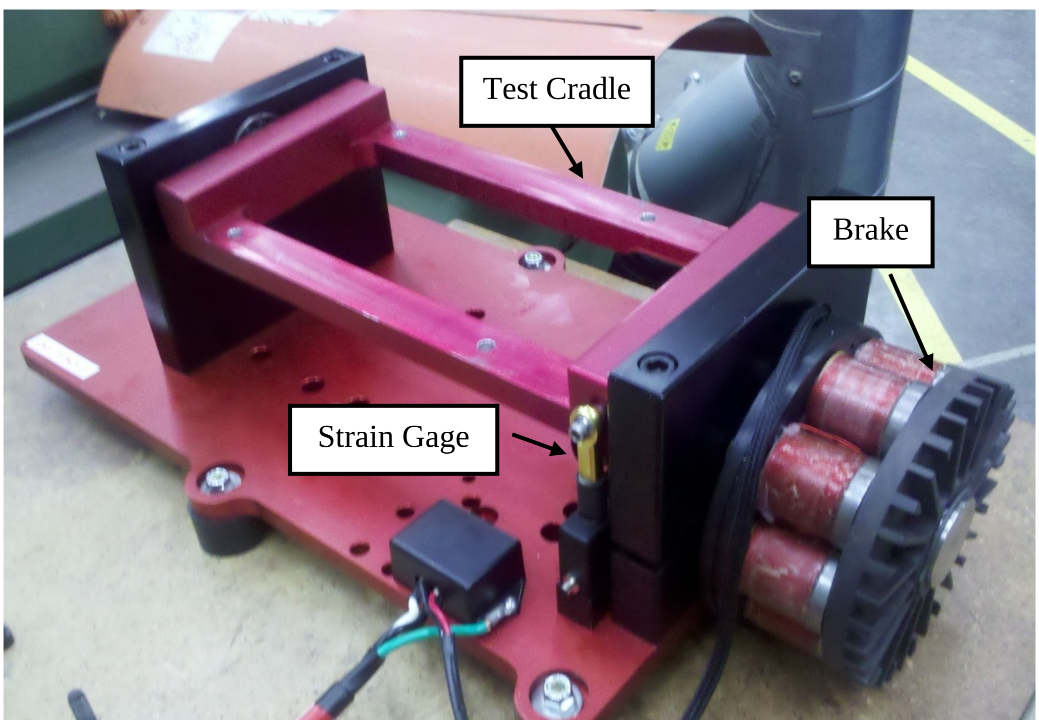


Figure 45: Land and Sea Dynamometer

6.2. Electric Motor

The Maxon RE-50 Brushed DC electric motor was selected for its controllability as discussed in Greiser [5]. It was used to validate the analysis and the results are discussed in the next chapter. The specific order number 370354, as shown in Figure 46, is rated for 200 W of continuous power. The power required for endurance is 124.2 W as listed in Table 3.

Figure 46: Maxon RE-50 Brushed DC Electric Motor

Figure 47 shows the proposed H-EPS setup with the Maxon motor in parallel with a Fuji BF25-EI engine. The EM is attached to the engine shaft through a one-way bearing mounted in the larger diameter gear. When the engine shaft is rotating, so is the EM shaft but, the EM shaft is allowed to rotate while the engine is idling or off. Mengistu also tested a Honda GX35 ICE. The specific engine testing data is discussed in Mengistu [44].

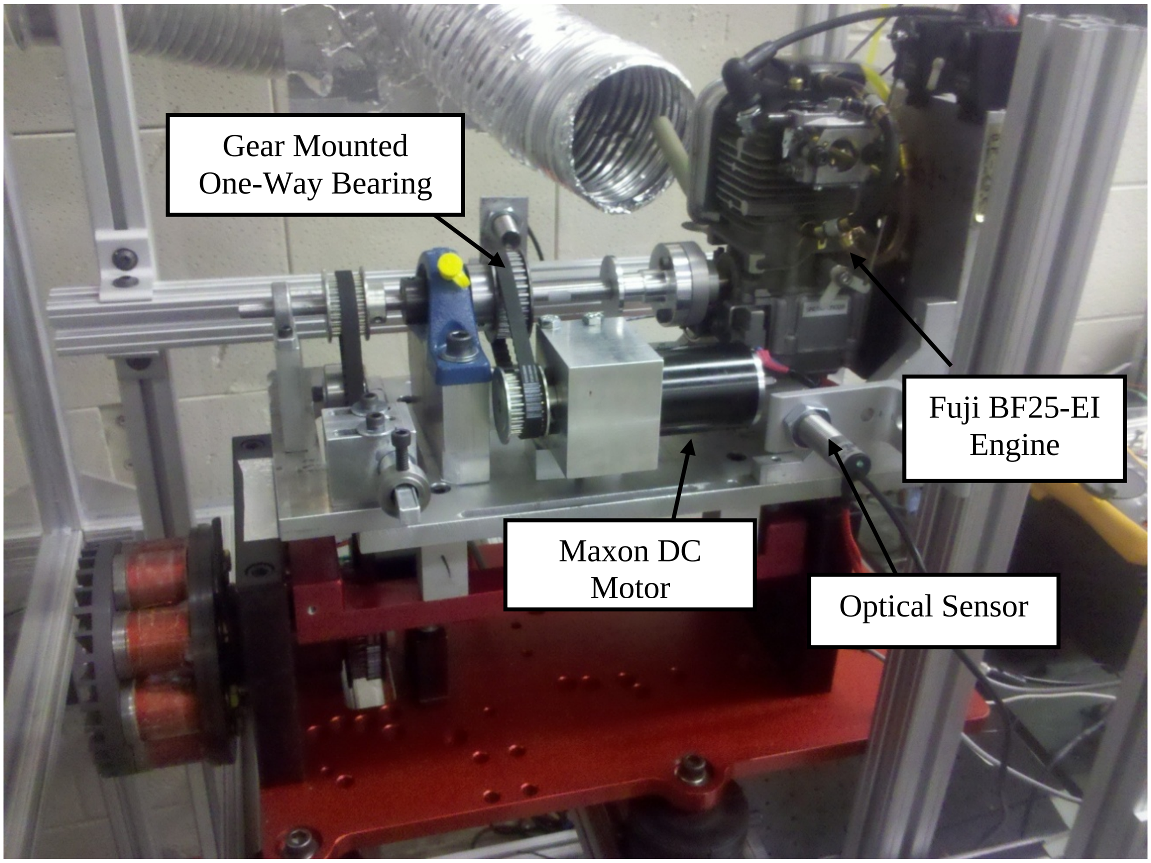


Figure 47: Testing Proposed H-EPS Setup

IV. Analysis and Results

1. Chapter Overview

This chapter discusses the results of the different analysis tool options. Although the tool could be used with any performance requirements, it was run to satisfy the propulsion system requirements specified by Hiserote [4]. Those requirements are listed in Table 3. The power required curve is shown in Figure 1.

Table 3: Proposed Mission Requirements [45]

Mission Segment	Power Required (W)	Velocity Required (kts)
Theoretical Endurance	87.4	18.0
Stall	96.8	23.0
Actual Endurance	124.2	28.0
Climb	367.9	28.0
Cruise	265.7	40.0
Max Velocity	828.1	60.0

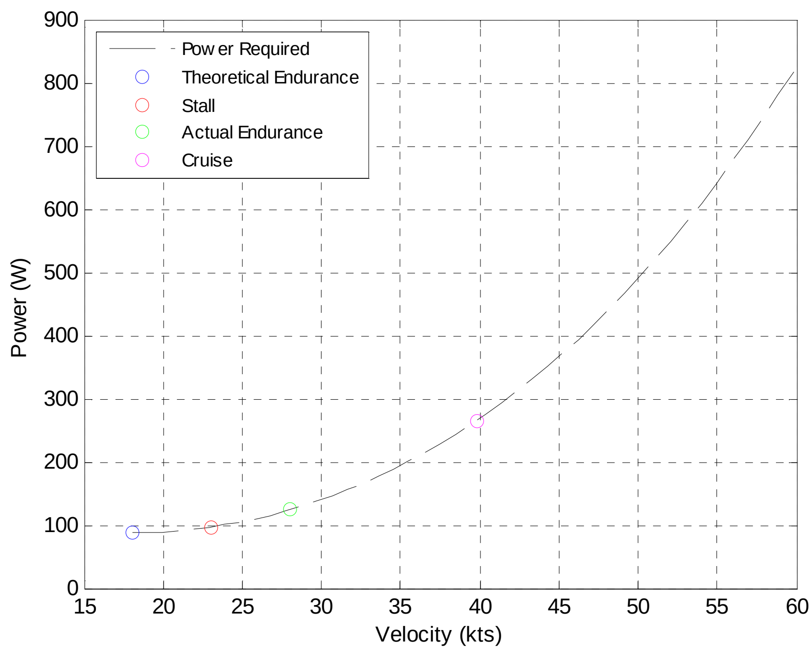


Figure 48: Proposed Power Required Curve [45]

1.1. Analytically Verify COTS Electric Motor Specifications

Table 4 lists the input variables that were taken from the manufacturer's specification sheet referenced in Table 2. From there, the analysis tool calculates the

Table 4: Manufacturer Specified Input Data

Parameter	Units	Manufacturer Data
Nominal Voltage	[V]	24
Max Continuous Current	[A]	9.15
EM Internal Resistance	[Ohms]	0.113
No-load Current	[A]	0.225

other parameters listed in Table 5. The analysis tool values were consistent with the provided manufacturer data, but the tool was able to determine much more. The manufacturer gave a nominal or maximum continuous current but did not list what the most efficient current was. The maximum efficiency, calculated at the most efficient current, was determined to be in line with what the manufacturer stated. The analysis tool also provided the optimum torque that produced the most efficient current as well as the input and output power. It was noted that the EM was rated at 200 W, but the most efficient output power was 155 W.

Table 5: Electric Motor Only Analysis vs. Manufacturer Data

Parameter	Units	Analysis Tool Output	Manufacturer Data
Max Continuous Torque	[N-m]	0.35	0.354
Nominal Speed	[rpm]	5595.75	5540
Optimum Current	[A]	6.91	
Starting Current	[A]	212.39	212
Stall Torque	[N-m]	8.40	8.420
Optimum Torque	[N-m]	0.27	
No-Load Speed	[rpm]	5784.00	5780
EM Input Power	[W]	165.90	
EM Output Power	[W]	155.28	
EM Efficiency	[%]	93.59	94

The Maxon motor efficiency map is shown is Figure 49 with the imposed constraints shown as colored lines. The magenta lines represent the voltage constraints, the red lines represent the rotational speed constraints and the green lines represent the current constraints. The area enclosed by the constraints is the feasible region. The active constraint is the maximum voltage. This is how the analysis tool determined the maximum efficiency. The power map is shown in Figure 50 and Figure 51 shows torque, power, and efficiency maps at various voltages. In some situations, the motor may not be operating at maximum voltage; therefore, it is important to understand how the EM operates at these different voltages.

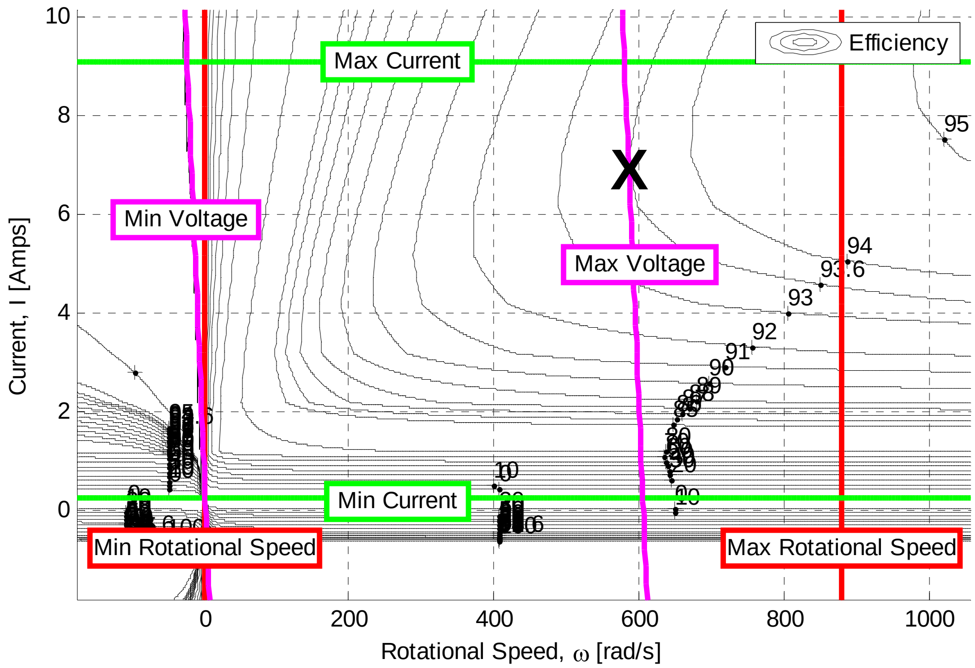


Figure 49: Maxon Motor Efficiency Map

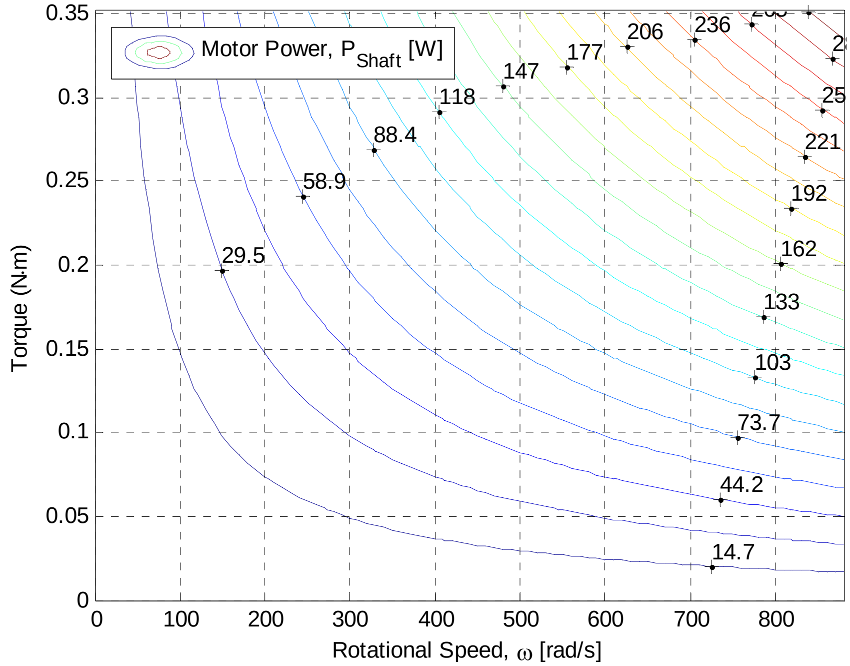


Figure 50: Maxon Motor Power Map

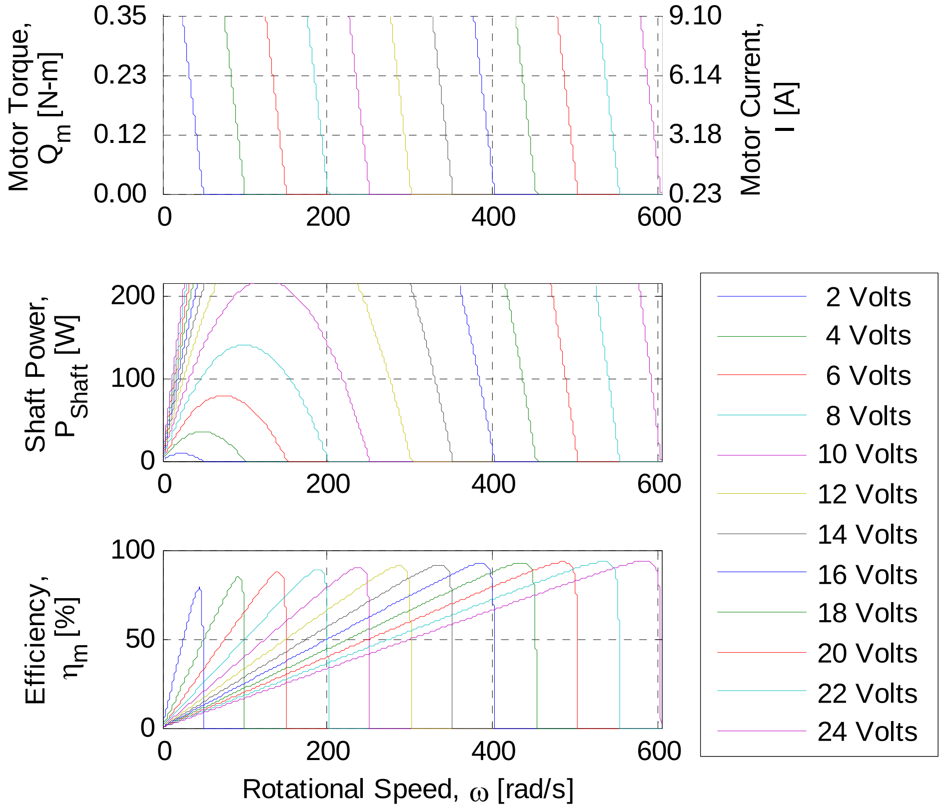


Figure 51: Maxon Motor Torque, Power, and Efficiency at Various Voltages

1.2. Experimental Results

As previously discussed, the experimental setup shown in Figure 47 was used to validate the results of the analysis tool. The setup involved testing the Maxon motor with 16:11 gear ratio. A DC power supply provided power to the EM through Greiser's controller. The EM was run at the required rotational speed as shown in Table 14. The dynamometer was used to imitate the predicted load the propeller would see at the endurance conditions. Once the test was on conditions, the voltage and current were read from independent multimeters while the torque and speed were read off the dynamometer display screen. The test was run seven times and the results are shown in Table 6 .

Table 6: Predicted vs. Experimental Results

Parameter	Units	Ideal	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Mean
Current	[A]	7.24	7.28	7.27	7.19	7.22	7.17	7.34	7.29	7.25
Voltage	[v]	23.82	24.01	24.14	24.7	24.72	23.78	23.78	23.78	24.13
Speed	[rpm]	5545	5350	5350	5450	5400	5300	5200	5250	5329
Torque	[N-m]	0.278	0.27	0.27	0.27	0.27	0.27	0.27	0.27	0.27
Efficiency	[%]	93.56	86.54	86.19	86.77	85.55	87.89	84.23	85.63	86.1

The efficiency was estimated to be 94 percent. The dynamometer test showed an average efficiency of only 86 percent. There are several possible reasons for the discrepancies. The biggest source of error is attributed to the crude test procedure. The dynamometer was used to measure torque and a separate optical sensor was used to measure rotational speed. This alleviated the need for determining the rotational speed through the gear ratio. Both the rotational speed and torque were time-varying and displayed as both a dial indicator and a numeric readout. These values were not constant and the throttle and load knob had to be tweaked until they were somewhat close to the specified rotational speed and torque. At the same time, the voltage and current were read off of independent multimeters. Again, they were time-varying and a judgment call was made as to what number they were bouncing around. Any of the four necessary parameters could have been read wrong.

A separate calculation was run where the measured rotational speed and torque were run through the analysis tool as if they were the propeller rotational speed and torque. Those results in Table 7 show that even if the rotational speed and torque were measured accurately, a small variation in the current and voltage readings could have major effects on the calculated efficiency. Separate tests were not possible due to issues with the dynamometer setup.

Table 7: Experimental Data Rerun Through Analysis Tool

Parameter	Units	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Mean
Speed	[rpm]	5350	5350	5450	5400	5300	5200	5250	5329
Torque	[N-m]	0.27	0.27	0.27	0.27	0.27	0.27	0.27	0.27
Current	[A]	7.15	7.15	7.15	7.15	7.15	7.15	7.15	7.15
Voltage	[v]	22.64	22.64	23.05	22.84	22.44	22.03	22.23	22.56
Efficiency	[%]	93.40	93.40	93.46	93.45	93.37	93.30	93.34	93.38

1.3. Hybrid Configuration

The hybrid configuration analysis tool was run the following four ways:

1. Maxon motor and propeller with the gear ratio equal to one
2. Maxon motor and propeller with optimized gear ratio
3. Maxon motor and propeller with selected gear ratio
4. Design optimized electric motor to the propeller

The first case is ideal for an in-line parallel hybrid configuration where the EM is mounted directly in line with the engine shaft as shown in Figure 52. If it is not possible for the EM to be mounted directly in line with the engine, then it would be mounted in the non in-line configuration as shown in Figure 53. If the non in-line configuration is chosen, then analysis tool is used to determine the best gear ratio between the propeller and EM to achieve the greatest EM efficiency. Because of how gear ratios are put together, the optimized gear ratio may not be available. Therefore, the analysis allows for the user to input a selected gear ratio. Finally, the analysis tool is used to design an EM for endurance based on a selected propeller.

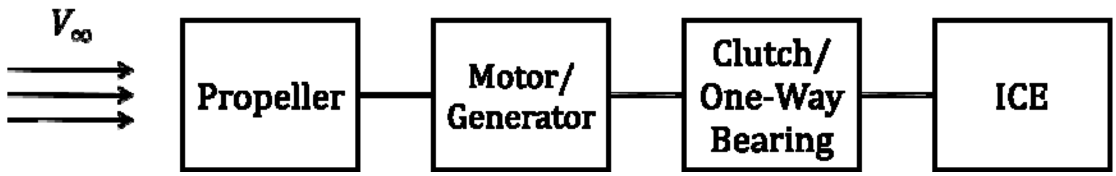


Figure 52: In-Line Parallel Hybrid Configuration

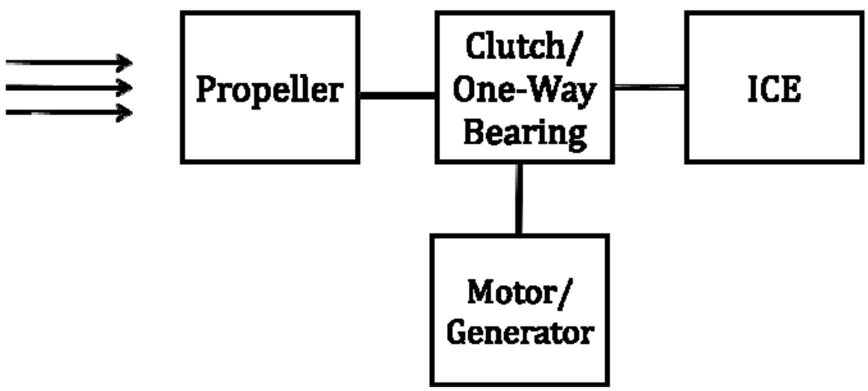


Figure 53: Non In-Line Parallel Hybrid Configuration

The above cases were run with experimental APC propeller data from Wichita State University. The five propellers are listed in Table 8 along with their required rotational speed and torque necessary to satisfy the endurance requirement.

Table 8: Required Propeller Rotational Speed and Torque for Endurance

Propeller Name	Rotational Speed [rad/s]	Torque [N-m]
APC 12x12	553.72	0.34
APC 18x8	399.19	0.40
APC 18x12	336.45	0.46
APC 20x10	319.66	0.49
APC 20x12	299.87	0.59

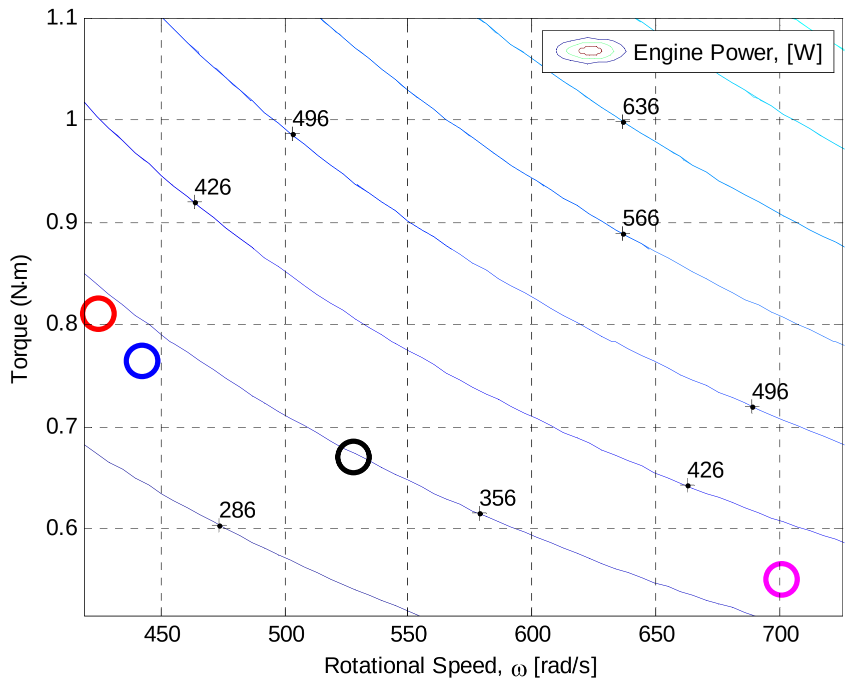
Table 9 lists the engine and propeller parameters for cruise. The required engine rotational speed and torque were calculated based on the propeller rotational speed and torque. Since there is not a gear ratio between the ICE and propeller, the rotational speeds are matched. The difference in the torque values is attributed to the assumed 99 percent efficient clutch/one-way bearing. The APC 18x12 propeller shows the highest efficiency for the cruise requirements.

Table 9: Engine and APC Propellers Operating at Cruise

Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
Cruise Speed	[m/s]	20.578	20.578	20.578	20.578	20.578
ICE Rotational Speed	[rad/s]	700.50	527.70	442.30	424.55	398.49
ICE Output Power	[W]	386.30	353.44	338.39	344.27	342.50
ICE Output Torque	[N-m]	0.55	0.67	0.77	0.81	0.86
Shaft Torque	[N-m]	0.55	0.66	0.76	0.80	0.85
Shaft Rotational Speed	[rad/s]	700.50	527.70	442.30	424.55	398.49
Shaft Power	[W]	382.44	349.91	335.01	340.83	339.07
Prop Rotational Speed	[rad/s]	700.50	527.70	442.30	424.55	398.49
Advance Ratio	[]	0.61	0.54	0.64	0.60	0.64
Prop Thrust	[N]	12.91	12.91	12.91	12.91	12.91
Prop Power	[W]	265.70	265.70	265.70	265.70	265.70
Prop Torque	[N-m]	0.55	0.66	0.76	0.80	0.85
Prop Efficiency	[%]	69.48	75.93	79.31	77.96	78.36

Figure 54 shows the required cruise rotational speed and torque for the APC propellers plotted on the Honda GX35 power map. The magenta circle is for the APC 12x12 propeller, the black circle is for the APC 18x8 propeller, the blue circle is for the

APC 18x12 propeller, and the red circle is for the APC 20x10 propeller. The APC 20x12 propeller is not shown because its required rotational speed is less than the Honda's minimum rotational speed. The map was produced with data collected by Mengistu [44].



Note: The colored circles from left to right are for the APC 20x10 (red), APC 18x12 (blue), APC 18x8 (black), and APC 12x12 (magenta).

Figure 54: Honda GX35 Power Map and APC Propellers Operating at Cruise

When the analysis tool is first run, the user is prompted to enter takeoff and mission altitudes. A standard atmosphere table is used to calculate the mission altitude density which is used to calculate the propeller parameters as shown in section II.3.2.

```
Enter takeoff altitude (meters AMSL):  0
Enter mission altitude (meters AGL):  305

Mission Altitude Density (kg/m^3) = 1.1895
```

Figure 55: Takeoff and Mission Altitude Input

1.3.1. Maxon Motor and Propeller with Gear Ratio Equal to One

Table 10 lists the output parameters from the analysis tool for the Maxon motor paired with five different APC propellers optimized for endurance with a gear ratio equal to one. The APC 12x12 propeller is the only one that would operate within the max continuous current limit of 9.15 A as specified by the manufacturer. The other propellers require more torque than the Maxon motor can deliver without a gear ratio. Since the EM

Table 10: Maxon Motor and APC Propellers with Gear Ratio Equal to One Optimized for Endurance

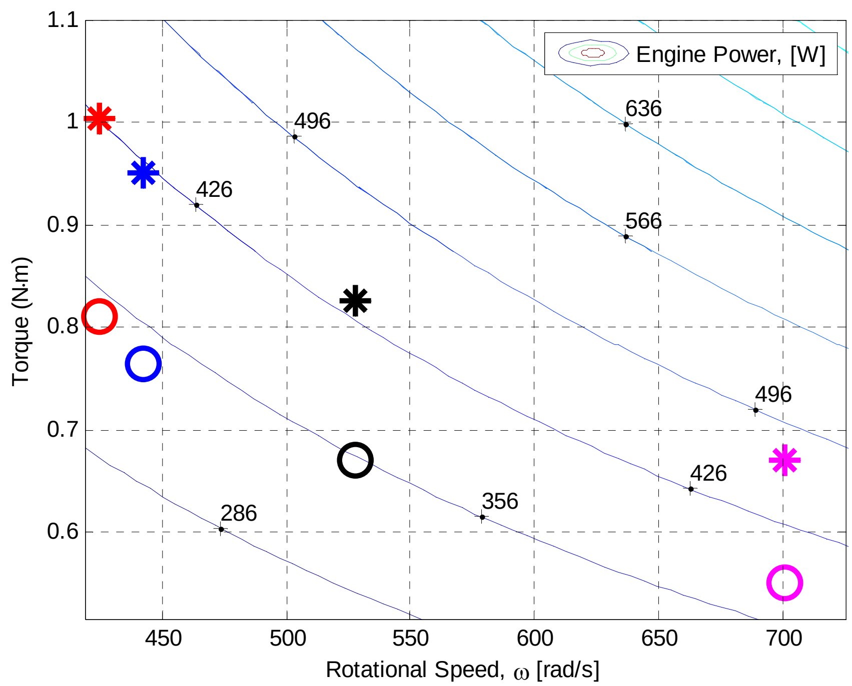
Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
Endurance Speed	[m/s]	14.40	14.40	14.40	14.40	14.40
Endurance Time	[hr]	1.87	1.58	1.38	1.31	1.23
EM Rotational Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Required Voltage (*V*)	[Volts]	22.94	17.00	14.68	14.09	13.39
Required Current (*I*)	[Amps]	8.82	10.43	11.96	12.58	13.37
Starting Current	[Amps]	202.99	150.41	129.94	124.67	118.52
Most Efficient Current at *V*	[Amps]	6.76	5.82	5.41	5.30	5.16
Most Efficient EM Speed at *V*	[rad/s]	559.61	412.35	355.13	340.44	323.27
Stall Torque	[N-m]	8.03	5.95	5.14	4.93	4.69
EM Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
No-Load Speed	[rad/s]	578.88	428.94	370.55	355.55	337.99
EM Input Power	[W]	202.42	177.26	175.57	177.29	179.02
EM Output Power	[W]	188.68	161.41	156.41	156.55	156.16
EM Efficiency	[%]	93.21	91.06	89.09	88.30	87.23
Max EM Efficient at *V*	[%]	93.45	92.41	91.85	91.68	91.48
Max Shaft Torque	[N-m]	1.18	1.18	1.18	1.18	1.18
Shaft Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
Shaft Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Shaft Power	[W]	188.68	161.41	156.41	156.55	156.16
Propeller Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Propeller Advance Ratio	[]	0.54	0.50	0.59	0.56	0.59
Propeller Thrust	[N]	8.62	8.62	8.62	8.62	8.62
Propeller Power	[W]	124.20	124.20	124.20	124.20	124.20
Propeller Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
Propeller Efficiency	[%]	65.83	76.95	79.41	79.34	79.53
Total Endurance Efficiency	[%]	61.36	70.07	70.74	70.05	69.38

would be air-cooled by the propeller, the additional required current may be acceptable, but needs to be studied further.

Table 11 lists the ICE and generator requirements for charging the batteries

Table 11: Cruise with Regeneration Requirements for Gear Ratio Equal to One

Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
Generator Output Power	[W]	75	75	75	75	75
Generator Speed	[rad/s]	700.50	527.70	442.30	424.55	398.49
Generator Required Torque	[N-m]	0.12	0.15	0.18	0.19	0.20
Generator Input Power	[W]	82.09	81.22	81.14	81.18	81.29
Generator Output Voltage	[V]	27.45	20.50	17.03	16.30	15.23
Generator Output Current	[A]	2.73	3.66	4.40	4.60	4.92
Generator Efficiency	[%]	91.36	92.34	92.44	92.39	92.26
ICE Output Power	[W]	469.22	435.48	420.35	426.27	424.61
ICE Output Torque	[N-m]	0.67	0.83	0.95	1.00	1.07



Note: The colored symbols from left to right are for the APC 20x10 (red), APC 18x12 (blue), APC 18x8 (black), and APC 12x12 (magenta). The circles are for cruise and the asterisks are for cruse plus regeneration.

Figure 56: Honda GX35 Power Map with Cruise and Regeneration Requirements for Gear Ratio Equal to One

during cruise. The generator output power was specified as 75 W. For APC 12x12 propeller, because of the rotational speed during cruise, the generator is required to provide 27.5 V with an allowable current draw of 2.7 A. This equates to an additional torque of 0.1 N-m that the ICE needs to supply. While maintaining speed, the ICE output power is increased from 386 W to 469 W to provide the additional required torque as shown by the black asterisk in Figure 56.

1.3.2. Maxon Motor and Propeller with Optimized Gear Ratio

Table 12 lists the output parameters from the analysis tool for the Maxon motor paired with the five different APC propellers optimized for endurance. For this case, the gear ratio was also optimized. The APC 12x12 propeller optimized gear ratio is 1. For the other four propellers, the optimized solution maxed out the voltage and allowed the current to drop. The lowest current draw comes from the APC 20x12 prop. Its bigger size allows it rotate much slower but with greater torque than the other propellers. For that reason, it requires the greatest gear ratio of 1.95. Even with the slight differences in current draw, the endurance times for the four larger propellers are within minutes of each other. The EM efficiency for these four propellers is roughly 93.6 percent. Comparing Table 11 with Table 12 shows that the EM's efficiency was increased by adding a gear ratio. This confirms that electric motors operate more efficiently at higher rotational speeds.

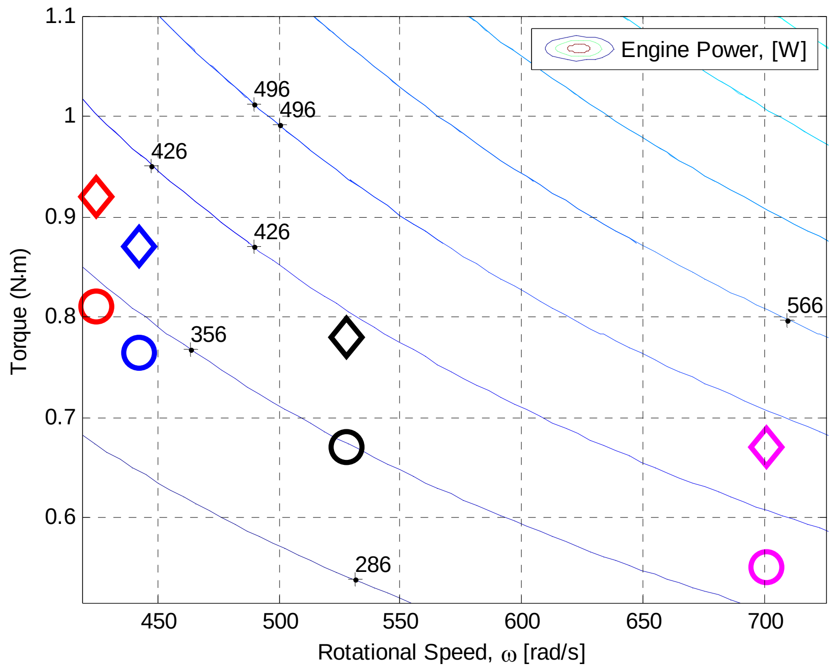
Table 12: Maxon Motor, APC Propellers, and Gear Ratio Optimized for Endurance

Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
Gear Ratio [EM/ICE]	[]	1	22/15	7/4	11/6	43/22
Gear Ratio [EM/ICE]	[]	1.00	1.47	1.74	1.83	1.95
Endurance Speed	[m/s]	14.40	14.40	14.40	14.40	14.40
Endurance Time	[hr]	1.87	2.30	2.37	2.37	2.37
EM Rotational Speed	[rad/s]	552.67	585.21	585.84	585.82	585.87
Required Voltage (*V*)	[Volts]	22.90	24.00	24.00	24.00	24.00
Required Current (*I*)	[Amps]	8.84	7.19	6.96	6.97	6.95
Starting Current	[Amps]	202.64	212.39	212.39	212.39	212.39
Most Efficient Current at *V*	[Amps]	6.75	6.91	6.91	6.91	6.91
Most Efficient EM Speed at *V*	[rad/s]	558.63	585.98	585.98	585.98	585.98
Stall Torque	[N-m]	8.02	8.41	8.41	8.41	8.41
EM Torque	[N-m]	0.34	0.28	0.27	0.27	0.27
No-Load Speed	[rad/s]	577.88	605.70	605.70	605.70	605.70
EM Input Power	[W]	202.44	172.46	167.11	167.26	166.84
EM Output Power	[W]	188.68	161.41	156.41	156.55	156.16
EM Efficiency	[%]	93.20	93.59	93.60	93.60	93.60
Max EM Efficient at *V*	[%]	93.45	93.60	93.60	93.60	93.60
Max Shaft Torque	[N-m]	0.35	0.52	0.62	0.65	0.69
Shaft Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
Shaft Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Shaft Power	[W]	188.68	161.41	156.41	156.55	156.16
Propeller Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Propeller Advance Ratio	[]	0.54	0.50	0.59	0.56	0.59
Propeller Thrust	[N]	8.62	8.62	8.62	8.62	8.62
Propeller Power	[W]	124.20	124.20	124.20	124.20	124.20
Propeller Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
Propeller Efficiency	[%]	65.83	76.95	79.41	79.34	79.53
Total Endurance Efficiency	[%]	61.35	72.02	74.32	74.26	74.44

Table 13 lists the ICE and generator requirements for charging the batteries during cruise for each of the five APC propellers. For all cases except the APC 12x12 propeller, the generators are operating at roughly the same voltage and current in order to supply the required 75 W of regeneration power. The additional torque required by the engine for each generator and propeller combination is represented by the different diamonds in Figure 57.

Table 13: Engine, Propeller, and Optimized Gear Ratio Combination Operating at Cruise with Regeneration

Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
Generator Output Power	[W]	75	75	75	75	75
Generator Speed	[rad/s]	699.18	773.59	770.14	778.06	778.56
Generator Required Torque	[N-m]	0.12	0.11	0.11	0.11	0.11
Generator Input Power	[W]	82.08	82.59	82.56	82.62	82.62
Generator Output Voltage	[V]	27.39	30.37	30.24	30.55	30.57
Generator Output Current	[A]	2.74	2.47	2.48	2.45	2.45
Generator Efficiency	[%]	91.37	90.81	90.84	90.78	90.78
ICE Output Power	[W]	469.21	436.86	421.79	427.72	425.96
ICE Output Torque	[N-m]	0.67	0.78	0.87	0.92	0.97



Note: The colored symbols from left to right are for the APC 20x10 (red), APC 18x12 (blue), APC 18x8 (black), and APC 12x12 (magenta). The circles are for cruise and the diamonds are for cruse plus regeneration.

Figure 57: Honda GX35 Power Map with Cruise and Regeneration Requirements with Optimized Gear Ratio

1.3.3. Maxon Motor and Propeller with Input Gear Ratio

Table 14 lists the output parameters from the analysis tool for the Maxon motor paired with three different APC propellers optimized for endurance. For this case, the gear ratio was set to 16:11. This gear ratio was chosen because it was the gear ratio that was needed for the H-EPS to incorporate the one-way bearing shown in Figure 47. The

Table 14: Maxon Motor and APC Propellers Optimized for Endurance with Selected Gear Ratio

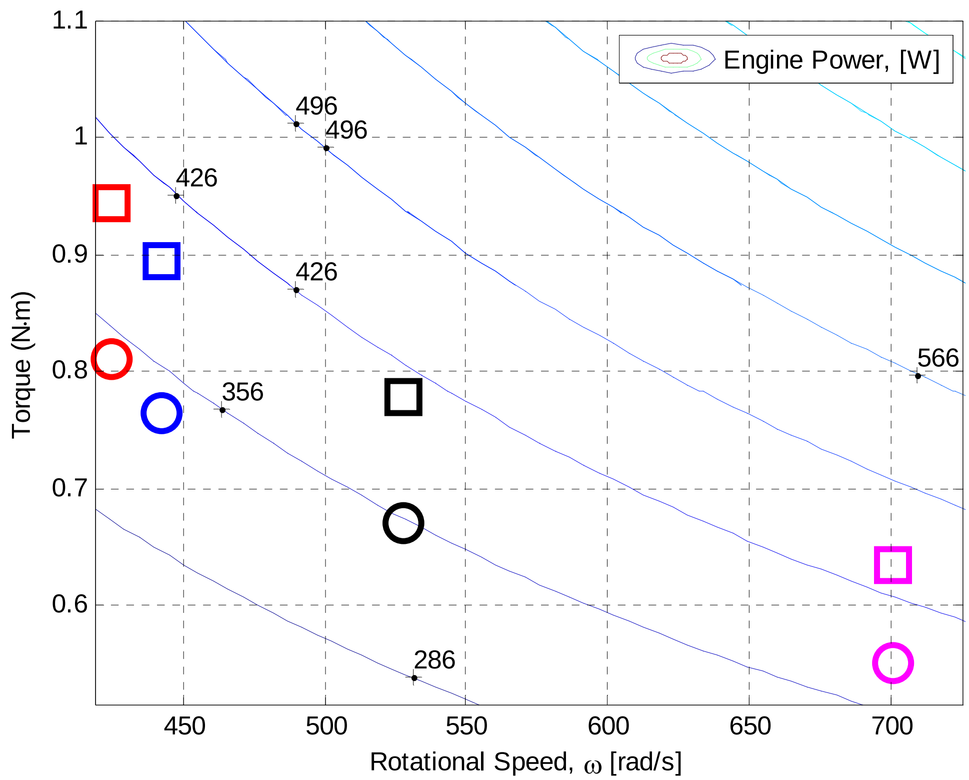
Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
Gear Ratio [EM/ICE]	[]	16/11	16/11	16/11	16/11	16/11
Gear Ratio [EM/ICE]	[]	1.4545	1.4545	1.4545	1.4545	1.4545
Endurance Speed	[m/s]	14.40	14.40	14.40	14.40	14.40
Endurance Time	[hr]	2.69	2.28	1.99	1.89	1.78
EM Rotational Speed	[rad/s]	805.40	580.64	489.39	464.96	436.18
Required Voltage (*V*)	[Volts]	32.61	23.83	20.33	19.41	18.33
Required Current (*I*)	[Amps]	6.14	7.24	8.29	8.72	9.26
Starting Current	[Amps]	288.55	210.84	179.89	171.76	162.21
Most Efficient Current at *V*	[Amps]	8.06	6.89	6.36	6.22	6.04
Most Efficient EM Speed at *V*	[rad/s]	799.93	581.65	494.89	472.11	445.36
Stall Torque	[N-m]	11.42	8.35	7.12	6.80	6.42
EM Torque	[N-m]	0.23	0.28	0.32	0.34	0.36
No-Load Speed	[rad/s]	822.91	601.29	513.03	489.83	462.58
EM Input Power	[W]	200.12	172.51	168.54	169.29	169.74
EM Output Power	[W]	188.68	161.41	156.41	156.55	156.16
EM Efficiency	[%]	94.29	93.57	92.80	92.47	92.00
Max EM Efficient at *V*	[%]	94.49	93.57	93.05	92.89	92.69
Max Shaft Torque	[N-m]	0.51	0.51	0.51	0.51	1.72
Shaft Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
Shaft Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Shaft Power	[W]	188.68	161.41	156.41	156.55	156.16
Propeller Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Propeller Advance Ratio	[]	0.54	0.50	0.59	0.56	0.59
Propeller Thrust	[N]	8.62	8.62	8.62	8.62	8.62
Propeller Power	[W]	124.20	124.20	124.20	124.20	124.20
Propeller Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
Propeller Efficiency	[%]	65.83	76.95	79.41	79.34	79.53
Total Endurance Efficiency	[%]	62.06	72.00	73.69	73.37	73.17

APC 12x12 propeller cannot operate with this gear ratio because the propeller was unable to produce the required power with the voltage limited to 24 V. Table 14 shows the EM would have to be operated at 32.61V to meet the power demand. The APC 20x12 propeller operating with this gear ratio will required more current then the manufacturer's specified maximum continuous current. As previously stated, this may not be an issue, but needs to be studied further. Based on the EM's maximum continuous current, the minimum gear ratio for the APC 20x12 propeller was determined to be 1.4725 and 16:11 is 1.4545. Again, the Honda's idol speed is still greater than the required rotational speed of the APC 20x12 propeller, so it cannot be used anyway. The APC 18x8 propeller provides the highest EM efficiency of the useable APC propellers with this gear ratio at the endurance condition. I also provides the longest endurance time.

Table 15 lists the ICE and generator requirements for charging the batteries during cruise for each of the five propellers. There is a clear difference in the operating current and voltage for the three useable propellers. The APC 18x8 propeller produces the lowest generator efficiency but the highest charging voltage while the APC 20x10 propeller produces the highest generator efficiency and the lowest charging voltage; maybe too low. The APC 18x12 rests comfortably between the two. The additional torque required by the engine for each generator and propeller combination is represented by the different squares in Figure 58.

Table 15: Engine, Propeller, and Selected Gear Ratio Combination Operating at Cruise with Regeneration

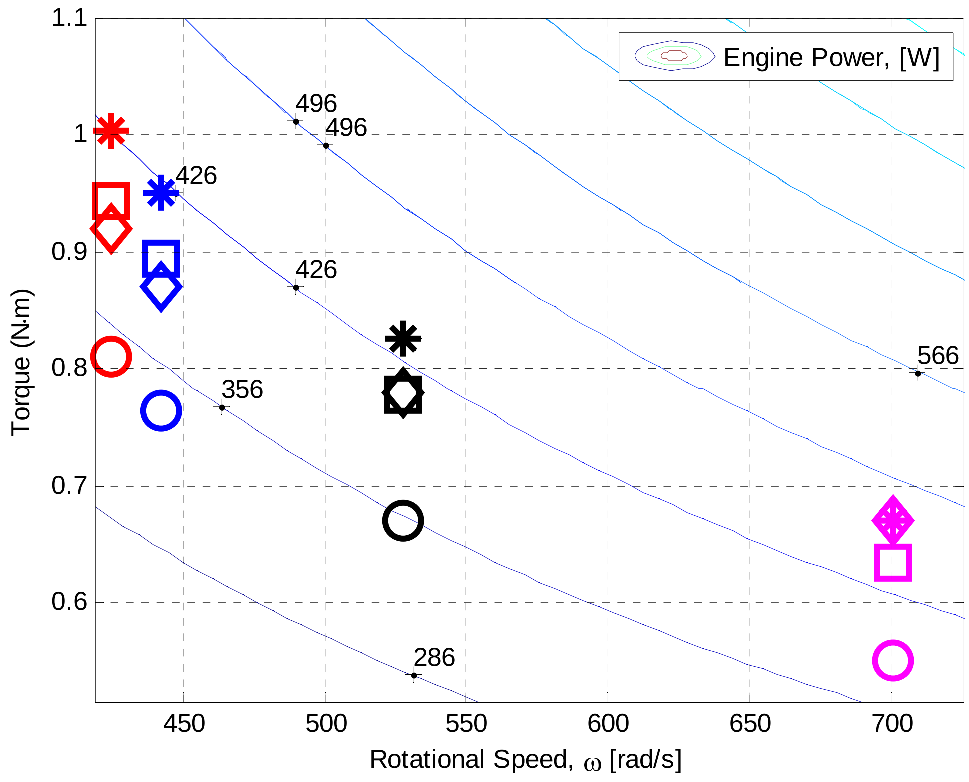
Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
Generator Output Power	[W]	75	75	75	75	75
Generator Speed	[rad/s]	1018.91	767.56	643.34	617.53	579.63
Generator Required Torque	[N-m]	0.08	0.11	0.13	0.13	0.14
Generator Input Power	[W]	84.48	82.54	81.74	81.60	81.41
Generator Output Voltage	[V]	40.16	30.13	25.15	24.12	22.59
Generator Output Current	[A]	1.87	2.49	2.98	3.11	3.32
Generator Efficiency	[%]	88.78	90.86	91.75	91.91	92.12
ICE Output Power	[W]	471.63	436.82	420.96	426.69	424.74
ICE Output Torque	[N-m]	0.64	0.78	0.89	0.94	1.00



Note: The colored symbols from left to right are for the APC 20x10 (red), APC 18x12 (blue), APC 18x8 (black), and APC 12x12 (magenta). The circles are for cruise and the squares are for cruse plus regeneration.

Figure 58: Honda GX35 Power Map with Cruise and Regeneration Requirements with Selected Gear Ratio

Figure 59 show the relationship between the cruise plus regeneration for each of the different gear ratio cases. The magenta asterisk and diamond represent the same point, because the optimized gear ratio between the Maxon motor and the APC 12x12 propeller is equal to 1 for the endurance condition. The black diamond and the square represent the same point because the gear ratio selected utilizing the one-way bearing was very close to the optimized gear ratio between the Maxon motor and the APC 18x8 propeller operating at the endurance condition.



Note: The colored symbols from left to right are for the APC 20x10 (red), APC 18x12 (blue), APC 18x8 (black), and APC 12x12 (magenta). The circles are for cruise. The asterisks are for cruse plus regeneration with a gear ratio equal to one. The diamonds are for cruse plus regeneration with an optimized gear ratio and the squares are for cruise plus regeneration with a selected gear ratio of 16:11.

Figure 59: Honda GX35 Power Map with Cruise and Regeneration Requirements

1.3.4. Design Optimized Electric Motor to a Propeller for Endurance

The original Maxon motor was constrained by its own nominal voltage and current. The designed EM's were constrained by the maximum system voltage and current. These maximums were based on the maximum output voltage and current of the DC-DC convertor used in the H-EPS. The DC-DC convertor allows the voltage to be stepped up from the installed battery packs. This in turn allows the current to be lowered for the same amount of power.

The analysis tool was run to design an optimized EM for each of the five propellers. The output key EM parameters are shown in Table 16 according to propeller size. In each case, the optimum solution was determined at the maximum voltage. Table 17 lists the output parameters from the analysis tool for each of the APC propellers paired with its designed EM optimized for endurance.

Table 16: Designed Electric Motor Parameters Based on Propeller Choice

Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
EM Internal Resistance	[Ohms]	0.228	0.276	0.287	0.286	0.288
No-load Current	[Amps]	0.243	0.217	0.211	0.212	0.211
Motor Speed Constant	[rpm/V]	136.148	98.268	82.845	78.704	73.842

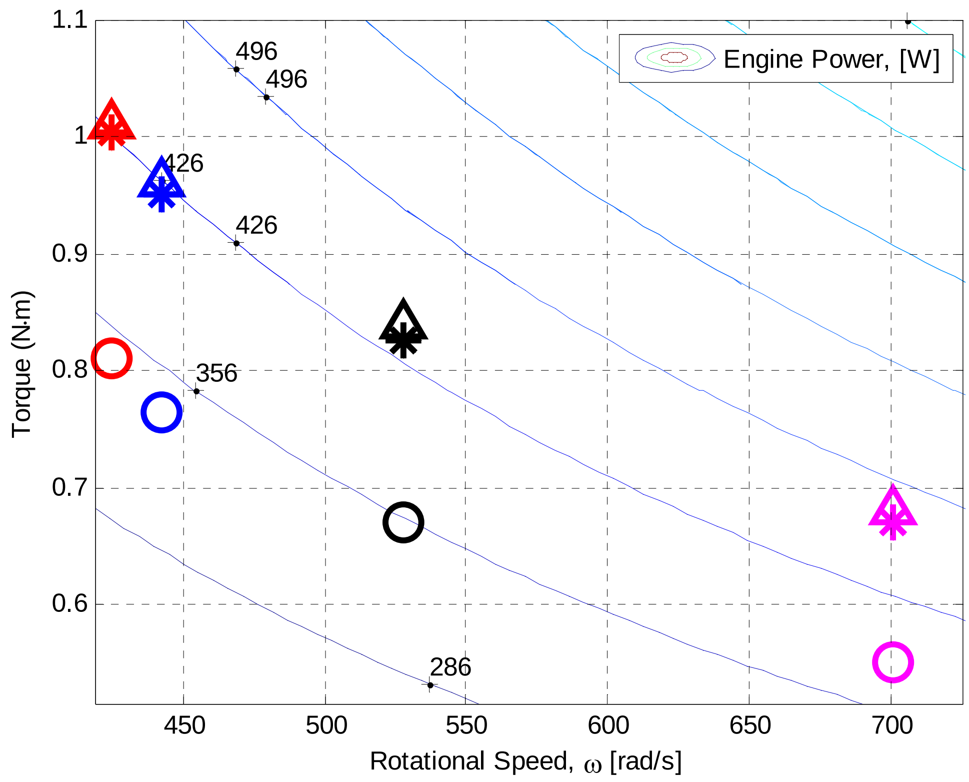
Table 18 lists the ICE and generator requirements for charging the batteries during cruise for each of the five propellers paired with their optimized electric motors. The generator output voltage and current for all five propellers are very close to each other with a voltage between 50 and 54 V and a current between 1.4 and 1.5 A. The generator efficiency for each pair is roughly 86 percent. The additional torque required by the engine for each generator and propeller combination is represented by the different triangles in Figure 60. The Figure also shows the relationship between the required torque for cruise plus regeneration for both the Maxon motor with a gear ratio equal to one and the optimized electric motors. It is interesting to note that for each electric motor and propeller combination, these points are very close to one another.

Table 17: Designed Electric Motor Endurance Parameters Based on Propeller Choice

Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
Endurance Speed	[m/s]	14.40	14.40	14.40	14.40	14.40
Endurance Time	[hr]	3.23	3.77	3.89	3.88	3.89
EM Rotational Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Required Voltage (*V*)	[Volts]	40.00	40.00	40.00	40.00	40.00
Required Current (*I*)	[Amps]	5.10	4.38	4.24	4.25	4.24
Starting Current	[Amps]	175.44	144.93	139.37	139.86	138.89
Most Efficient Current at *V*	[Amps]	6.53	5.60	5.43	5.44	5.41
Most Efficient EM Speed at *V*	[rad/s]	549.08	395.72	333.50	316.84	297.25
Stall Torque	[N-m]	12.29	14.06	16.04	16.94	17.93
EM Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
No-Load Speed	[rad/s]	570.30	411.63	347.02	329.67	309.31
EM Input Power	[W]	204.04	175.10	169.78	169.93	169.51
EM Output Power	[W]	188.68	161.41	156.41	156.55	156.16
EM Efficiency	[%]	92.47	92.18	92.12	92.13	92.12
Max EM Efficient at *V*	[%]	92.70	92.42	92.36	92.37	92.36
Max Shaft Torque	[N-m]	2.09	2.89	3.43	3.61	3.85
Shaft Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
Shaft Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Shaft Power	[W]	188.68	161.41	156.41	156.55	156.16
Propeller Speed	[rad/s]	553.72	399.19	336.45	319.66	299.87
Propeller Advance Ratio	[]	0.54	0.50	0.59	0.56	0.59
Propeller Thrust	[N]	8.62	8.62	8.62	8.62	8.62
Propeller Power	[W]	124.20	124.20	124.20	124.20	124.20
Propeller Torque	[N-m]	0.34	0.40	0.46	0.49	0.52
Propeller Efficiency	[%]	65.83	76.95	79.41	79.34	79.53
Total Endurance Efficiency	[%]	60.87	70.93	73.15	73.09	73.27

Table 18: Engine, Propeller, and Optimized EM Combination Operating at Cruise with Regeneration

Parameter	Units	APC 12x12	APC 18x8	APC 18x12	APC 20x10	APC 20x12
Generator Output Power	[W]	75.00	75.00	75.00	75.00	75.00
Generator Speed	[rad/s]	726.74	555.08	464.52	447.13	419.72
Generator Required Torque	[N-m]	0.12	0.16	0.19	0.19	0.21
Generator Input Power	[W]	87.88	87.22	86.90	87.05	87.01
Generator Output Voltage	[V]	50.63	53.55	53.14	53.85	53.88
Generator Output Current	[A]	1.48	1.40	1.41	1.39	1.39
Generator Efficiency	[%]	85.35	85.99	86.31	86.16	86.19
ICE Output Power	[W]	519.24	490.47	471.22	480.25	477.54
ICE Output Torque	[N-m]	0.71	0.88	1.01	1.07	1.14



Note: The colored symbols from left to right are for the APC 20x10 (red), APC 18x12 (blue), APC 18x8 (black), and APC 12x12 (magenta). The circles are for cruise. The asterisks are for cruse plus regeneration with a gear ratio equal to one for the Maxon Motor. The triangles are for cruise plus regeneration with the propellers optimized electric motor.

Figure 60: Honda GX35 Power Map with Cruise and Regeneration Requirements for the Maxon motor and Optimized Electric Motor Propeller Combinations

1.4. Well-Matched System

Figure 61 graphically shows the results of the Maxon motor matched to the APC 18x8 propeller with a 16:11 gear ratio for endurance. The top figure shows the propeller's thrust as a function of rotation speed for a given velocity at endurance. From the required thrust, the propeller's required rotational speed is determined.

Moving down to the second figure shows that the propeller's torque curve crosses the EM torque curve or in this case, the shaft's torque curve, due to the gear ratio. The fact that shaft and propeller torques are equal is what defines a matched system. This point is what defines the EM required voltage and current.

Moving down to the bottom two graphs, it is shown that both components are operating near their max efficiencies and therefore the system is well-matched. The difference in the rotational speed axes of the EM and the propeller is attributed to the gear ratio. This EM and propeller combination is well suited for the proposed H-EPS.

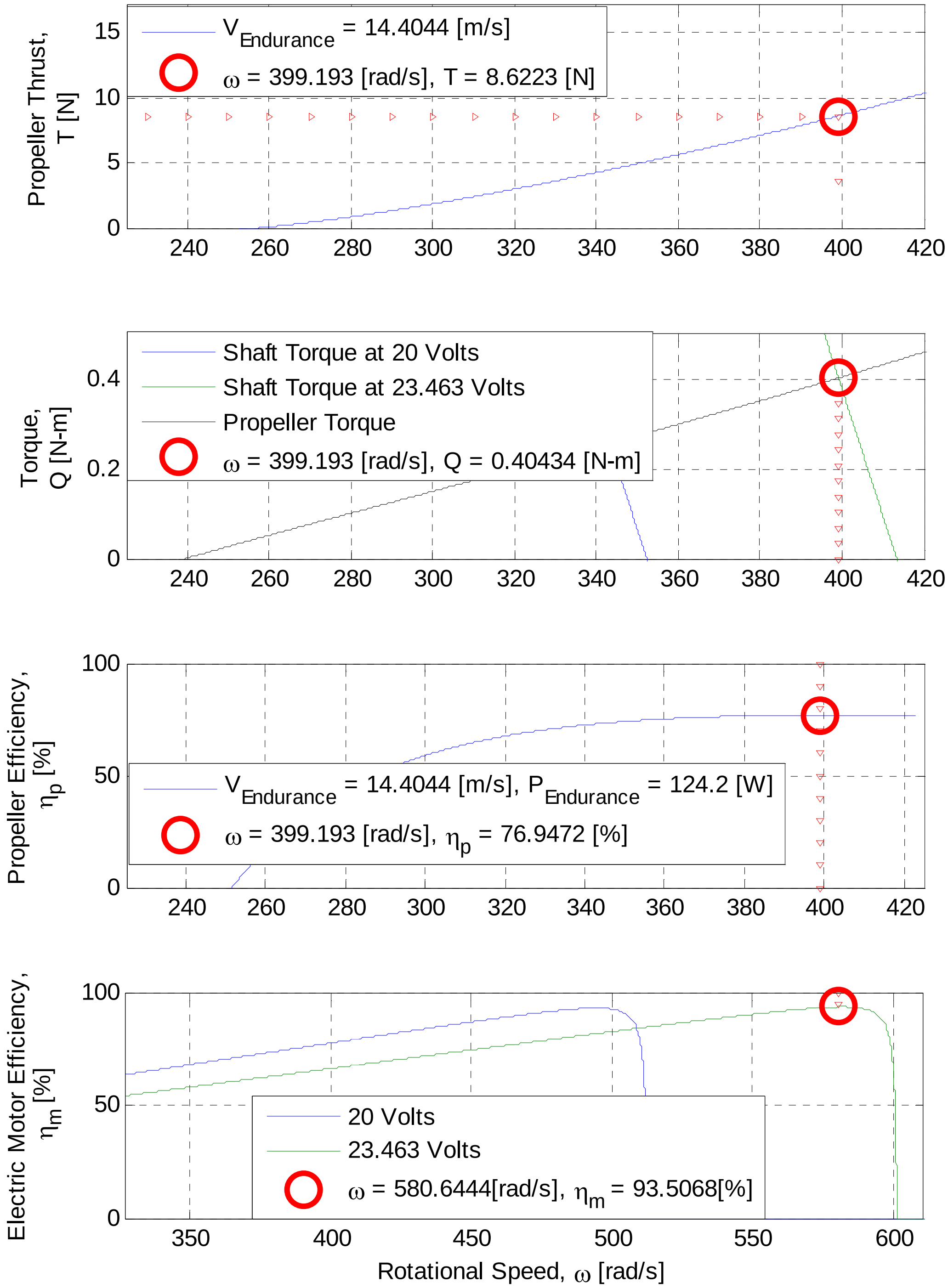


Figure 61: Maxon Motor Match with APC 18x8 Propeller and 16/11 Gear Ratio

V. Conclusions and Recommendations

1. Conclusions of Research

As the use of RPA continues to grow throughout the Department of Defense and many other government and civilian agencies, so does the interest in making them more efficient, economically friendly, quiet, and increasing their endurance time. Utilizing H-EPS provide these enhanced capabilities, but there have not been useful tools for matching or optimizing the system components.

The author has developed an analysis tool that takes simple electric motor parameters that, if not provided by the manufacturer, can be determined and paired with experimental propeller data to optimize the endurance phase of a typical RPA flight profile. The analysis tool also provides the rotational speed and torque requirements the engine needs to provide to satisfy the cruise phase of the flight profile with the option of recharging the system batteries.

This thesis has demonstrated the analysis tool's capability to match the electric motor and propeller with and without a gear ratio, to optimize the gear ratio for the greatest efficiency and endurance, as well as design the most optimal electric motor for a propeller based on performance requirements

2. Recommendations for Future Research

The analysis tool is not perfect. It was based on first order electric motor equations and does not take material make up or operating temperatures into account, which are critical in truly designing the best electric motor. Also more research needs to be conducted on the relationship between an electric motor's no-load current and internal resistance. Figure 29 plots three lines in a cloud of data to make a prediction of no-load current as a function of internal resistance. Choosing a different line yields completely different results.

The ability of the analysis tool to access a data base of experimental propeller data would greatly improve its analysis capability. Currently, the user is required to test each propeller independently and compare the results. Automating this process would be a significant improvement.

At the present time, the majority of user inputs are made within the MATLAB script file. Another much needed improvement is the design of a graphical interface where the user can make option selections and input desired information.

VI. Bibliography

[1] Laurence R. Newcome, *Unmanned Aviation: A Brief History of Unmanned Aerial Vehicles*. Reston, VA: AIAA, 2004.

[2] J. R. Wilson, "UAV Worldwide Roundup 2009," *Aerospace America*, vol. 47, no. 4, April 2009.

[3] Frederick G. Harmon, Andrew A. Frank, and Jean-Jaques Chattot, "Conceptual Deisgn and Simulation of a Small Hybrid-Electric Unmanned Aerial Vehicle," *Journal of Aircraft*, vol. 43, no. 5, October 2006.

[4] Ryan M. Hiserote, "Analysis of Hybrid-Electric Propulsion System Designs for Small Unmanned Aircraft Systems," Air Force Institute of Techonology, Wright-Patterson Air Force Base (WPAFB), Master's Thesis 2010.

[5] Collin Greiser, "Implementation of a Rule-Based Open-Loop Control Strategy for a Hybrid-Electric Propulsion System on a Small RPA," Air Force Institute of Technology, Wright-Patterson Air Force Base (WPAFB), Master's Thesis 2011.

[6] Peter Van den Bossche. (2010, May) Wikipedia. [Online]. http://en.wikipedia.org/wiki/File:Hybridpar.png

[7] Peter Van den Bossche. (2010, May) Wikipedia. [Online]. http://en.wikipedia.org/wiki/File:Hybridpeak.png

[8] Ernest H. Wakefield, *History of the Electric Automobile: Hybrid Electric Vehicles*. Warrendale, PA: Society of Automotive Engineers (SAE), 1998.

[9] (2009, March) 100th Anniversary of First US Hybrid Car Patent. [Online]. http://www.hybridcars.com/history/100th-anniversary-first-us-hybrid-car-patent-25616.html

[10 (2011, January) Lohner-Porsche Mixte Hybrid. [Online]. http://en.wikipedia.org/wiki/Lohner-Porsche_Mixte_Hybrid

[11 (2011) Porsche History: Milestones. [Online].

http://www.porsche.com/usa/aboutporsche/porschehistory/milestones/

[12 John M. Miller, *Propulsion Systems for Hybrid Vehicles*. London, UK: IEEE Power & Energy Series, 2004.

[13 Mehrdad Ehsani, Yimin Gao, and Ali Emadi, *Modern Electric, Hybrid Electric and, and Fuel Cell Vehicles: Fundamentals, Theory, and Design*, 2nd ed. New York, NY: CRC Press, 2010.

[14 Massimo Barcaro, Nicola Bianchi, and Silverio Bolognani, "Hybrid Electric Propulsion Systems Using Submersed SPM Machine," in *International Conference on Electrical Machines*, 2008.

[15 Richard R. Glassock, "Multimodal Hybrid Powerplant for Unmanned Aerial Systems (UAS) Robotics," Queensland University of Technology, Brisbane, Australia,.

[16 Jason Paur. (2009, July) Hybrid Power Comes to Aviation. [Online]. http://www.wired.com/autopia/2009/07/hybrid-aviation

[17 Ricahrd R. Glassock, Jane Y. Hung, Luis Felipe Gonzalez, and Rodney A. Walker, "Design, Modeling and Measurement of Hybrid Powerplant for Unmanned Aerial Systems (UAS)," in *5th Australian Congress on Applied Mechanics (ACAM)*, Brisbane, Australia, 2007.

[18 Thomas H. Bradely, Blake A. Moffitt, Thomas F. Fuller, Dimitri N. Mavris, and David E. Parekh, "Comparison of Design Methods for Fuel-Cell-Powered Unmanned Aerial Vehicles," *Journal of Aircraft*, vol. 46, no. 6, November-December 2009.

[19 Thomas H. Bradley, Blake A. Moffitt, David E. Parekh, Thomas F. Fuller, and Dimitri N. Marvis, "Energy Managemetn for Fuel Cell Powered Hybrid-Electric Aircraft," in *7th International Energy Conversion Engineering Conference*, Denver, CO, 2009.

[20 Blake A. Moffitt, Thomas H. Bradely, Dimitri Mavris, and David E. Perekh, "Design Space Exploration of Small-Scale PEM Fuel Cell Long Endurance Aircraft," in *6th AIAA Aviation Technology, Integration and Operations Conference (ATIO)*, Wichita, KS, 2006.

[21 Uche C. Ofoma and Chivey C. Wu, "Design of a Fuel Cell Powered UAV for

Environmental Research," in *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, Chicago, IL, 2004.

[22 Grant D. Rhoads, Nick A. Wagner, Brian J. Taylor, Derek B. Keen, and Thomas H. Bradley, "Design and Flight Test Results for a 24 Hour Fuell Cell Unmanned Aerial Vehicle," in *8th Annual International Energy Conversion Engineering Conference*, Nashville, TN, 2010.

[23 Dave Demerjian. (2008, April) Boeing Fuel Cell-Powered Plane Takes Flight. [Online]. http://www.wired.com/autopia/2008/04/in-an-aviation/#

[24 Irving M. Gottlieb, *Electric Motors and Control Techniques*, 2nd ed. New York, NY: TAB Books, 1994.

[25 Padmaraja Yedamale, "Brushless DC (BLDC) Motor Fundamentals," Microchip Technology Inc., AN885, 2003.

[26 Mark Drela, "First-Order DC Electric Motor Model," Massachusetts Instituted of Technology, February 2007.

[27 Mark Drela, "DC Motor / Propeller Matching," Massachusetts Institute of Technology, March 2005.

[28 D. Lundstrom, K. Amadori, and P. Krus, "Validation of Small Scale Electric Propulsion System Models," in *48th AIAA Aerospace Sciences Meeting*, Orlando, 2010.

[29 (2010, Jun) Wikipedia. [Online]. http://en.wikipedia.org/wiki/Joule_heating

[30 Northwestern University. (2010, December) Brushed DC Motor Theory. [Online]. http://hades.mech.northwestern.edu/index.php/Brushed_DC_Motor_Theory

[31 Robert J. Boucher, *The Electric Motor Handbook*, 2nd ed., 1995.

[32 John D. Anderson Jr., *Introduction to Flight*, 4th ed. Boston, MA: McGraw-Hill, 2000.

[33 Daniel P. Raymer, *Aircraft Desigh: A Conceptual approach*, 3rd ed. Reston, VA: American Institute of Aeronautics and Astronautics, 1999.

[34 Department of the Air Force; Department of the Navy; Department of Commerce, *ANC-9 Bulletin: Aircraft Propeller Handbook*, 1st ed.: U.S. Government Printing Office, 1956.

[35 Monal P. Merchant and L. Scott Miller, "Propeller Performance Measurements for Low Reynolds Number UAV Applications," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, 2006.

[36 Michael Ol, Cale Zeune, and Mike Logan, "Analytical-Experimental Comparison for Small Electric Unmanned Air Vehicle Propellers," in *26th AIAA Applied Aerodynamic Conference*, Honolulu, Hawaii, 2008.

[37 Dustin E. Gamble and Andrew Arean, "Automated Dynamic Propeller Testing at Low Reynolds Number," in *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, Orlando, Florida, 2010.

[38 John B. Brandt and Michael S. Selig, "Propeller Performance Data at Low Reynolds Numbers," in *49th AIAA Aerospace Sciences Meeting*, Orlando, Florida, 2011.

[39 John B. Heywood,. New York, NY: McGraw-Hill, 1988, ch. 1-2,6,15.

[40 Jasbir S. Arora, *Introduction to Optimum Design*, 2nd ed. San Diego, CA: Elseveir Academic Press, 2004.

[41 Ohad Gur and Aviv Rosen, "Optimizing Electric Propulsion Systems for Unmanned Aerial Vehicles," *Jounral of Aircraft*, vol. 46, no. 4, August 2009.

[42 Maxon Precision Motors Incorporated. (2010, May) Maxon Motors RE50 Spec Sheet. [Online]. https://downloads.maxonmotor.com/Katalog_neu/eshop/Downloads/Katalog_PDF/maxon_dc_motor/RE-programm/new/newpdf_10/RE-50-370354_10_EN_083.pdf

[43 Land and Sea Incorporated, *DYNOMite Owner's Manual*. Concord, NH, USA, 2006.

[44 Isseyas H. Mengistu, "Small Internal Combustion Engine Testing for Advanced Hybrid-Electric Remotely Piloted Aircraft Propulsion," Air Force Institute of Technology, Wright Patterson Air Force Base, M.S. Thesis 2011.

[45 Ryan M. Hiserote and Frederick G. Harmon, "Analysis of Hybrid-Electric Propulsion System Design for Small Unmanned Aircraft Systems," Air Force Research Laboratory, Wright Patterson AFB, AIAA 2010-6687, 2010.

Vita

Captain Todd A. Rotramel graduated from West High School in Wichita, KS in 1998. He completed his Bachelor of Science in Aerospace Engineering (B.S.E.) degree at Wichita State University, Wichita, KS, in 2003. He received his United States Air Force commission on December 17, 2004 after completing Officer Training School at Maxwell AFB, AL.

His first Air Force assignment was at the 412th Electronic Warfare Group at Edwards AFB, CA, where he served as an F-22 Electronic Warfare Test Engineer and the Executive Officer to the group commander. While stationed at Edwards AFB, Capt Rotramel was deployed in support of Operation Iraqi Freedom to the Coalition Air Force Transition Team (CAFTT) , Multi-National Security Transition Coalition-Iraq (MNSTC-I), Kirkuk Air Base, Iraq where he worked to rebuilding a fleet of Iraqi Air Force Comp Air 7SLX aircraft. In September 2009, he entered the Graduate School of Engineering and Management at the Air Force Institute of Technology. Upon completion of a Master's degree in Aeronautical Engineering in March 2010, he will be assigned to the Air Force Research Laboratory's Propulsion Directorate at Wright-Patterson AFB, OH.

Appendix A: MATLAB Code

```
function []=RPA_HEPS_Design()
% An analysis tool to size the components of a parallel hybrid-electric
% propulsion system using predicted endurance and curse parameteres.
% Capt Todd Rotramel (USAF)
% Air Force Institute Technology
% Master's Thesis: 'Optimization of Hybrid electric
% Propulsion System for Small Remotely-Piloted Aircraft'
%
% Last Updated: 26 Feb 2011
tic
% Clear Workspace
close all; clc; clear EM_Current_vs_volt EM_Torque_vs_volt
EM_Power_vs_volt
clear EM_Eff_vs_volt EEE EM_T_V EM_E_V

global R I0 wmax wmin Kvmax Kvmin Imax Imin Vmax Vmin Qmax Qmin GRmax
global GRmin I V Kv Pend Vend R_Range Pcruise Eff_Clutch PGen T a P
Q_EM
global Qmax_Shaft Qmin_EM Qmax_EM Q_Shaft_End Vemf Q_Stall w0 I_Stall
global I_maxEff w_maxEff Pin_EM Pout_EM tend w_EM N_EM N_maxEff N0
EM_Eff
global Effmax_EM w_Shaft_End P_Shaft_End w_Prop_End N_Prop_End
global Prop_J_End Prop_T_End Prop_P_End Prop_Q_End Prop_Eff_End
global w_Prop_Cruise N_Prop_Cruise P_ICE_Cruise Q_ICE_Cruise
Prop_Q_Cruise
global P_Shaft_Cruise Prop_J_Cruise Prop_T_Cruise Prop_P_Cruise
global Prop_Eff_Cruise Pout_Gen w_Gen Q_Gen Pin_Gen V_Gen I_Gen
global Eff_Gen P_ICE_Cruise_Gen Q_ICE_Cruise_Gen GR Test_Case EM_Case
global GR_Case ctrR ctrRmax
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%Analysis Tool
Options%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Test_Case=2;%      1=Electric Motor Only;       2=Hybrid Configuration
EM_Case=2;%        1=Test Electric Motor        2=Design Electric
Motor;
GR_Case=2;%        1=Optimized Gear Ratio       2=Input Gear Ratio
Plot_Switch=2;%    1=Plots;                     2=No Plots
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%Batteries
V_bat=25.9;%       [V]         Single Battery Pack Voltage
Num_bat=5;%        [#]         Number of Battery Packs
C_bat=3300;%       [mA/r]      Battery Capacity
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%DC-DC Convertor Output/System Maximum Voltage and Current
Imax=30;%        [A]         Max System Current
Vmax=40;%        [V]         Max System Voltage
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
% Gear Ratio
%%%Optimized Gear Ratio
if GR_Case==1%
    GRmax=20;%  [#] Max EM/Prop Shaft Gear Ratio
    GRmin=.25;%   [#] Min EM/Prop Shaft Gear Ratio
    GR=1;%      [#] DO NOT CHANGE Initial Condition
%%% Input Gear Ratio
elseif GR_Case==2
    EM_Gear_Teeth=22;%   [#]   Number of teeth on EM Gear
    Shaft_Gear_Teeth=32;%[#]   Number of teeth on Shaft Gear
    GR=Shaft_Gear_Teeth/EM_Gear_Teeth;%[#/1]   EM/Prop Shaft Gear Ratio
    GR=1;%               [#]   User input gear ratio or GR
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%Electric Motor for Endurance
Pend=124.2;%       [W]         Power Required for cruise
Vend=28;%          [knots]     Endurance Velocity 25-30 knots
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
% Electric motor internal resistance range for designing the optimal EM
R_Range=(.001:.001:3);%    [ohms]
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Initial_Plot_Switch=Plot_Switch; % used to get the specified user input
if EM_Case==1;% Test EM
    R_Range=1;
    ctrRmax=length(R_Range);
else% Design EM
    ctrRmax=length(R_Range)+1;
    Plot_Switch=2;%  Turn off the plots while the EM is being optimized
end               %  to save time

% Initialize the output matrix for designing the EM
Output_R_Range=zeros(length(R_Range),10);
```

```
%%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%

if Test_Case==1 && EM_Case==2
    disp('')
    disp('ERROR!')
    disp('You cannot test an electric motor that has not be designed.
');
    disp('First set Test_Case=2 and EM_Case==1.')
    disp('')
    disp('')
    disp('ERROR!')
    disp('')
    pause
end
ctrR=1;
for ctrR=1:ctrRmax
    if ctrR<length(R_Range) && ctrR>1
        close all% keeps the wait bar to one display
    end
    if ctrR<=length(R_Range)
        R=R_Range(ctrR);
        I0_Range=0.1./R_Range.^.6;%  [Amps] Estimation of Electric
Motor
        %                                   No-Load Current
        Kvmax_N=3500;%         [rpm/V] Max electric motor speed
constant
        Kvmin_N=50;%           [rpm/V] Min electric motor speed
constant

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        I0=I0_Range(ctrR);
        Imin=I0;
        if ctrR==1
            Qmin=.01;
            wmin=.01;
        end
        Qmax=Qmin;
        wmax=wmin;
        Kvmax=Kvmax_N*(2*pi/60);%      [(rad/s)/V]   Max EM speed
constant
        Kvmin=Kvmin_N*(2*pi/60);%      [(rad/s)/V]   Min EM speed
constant
        %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Know EM Specifications
    if EM_Case==1
        R=0.113;%          [ohms]      Electric Motor Internal
Resistance
        I0=0.225;%         [Amps]      Known Electric Motor No-Load
Current
        EM_NC=30;%        [V]         Known Electric Motor Nominal
Current
        EM_NV=40;%         [V]         Known Electric Motor Nominal
Voltage
        Kvmax_N=241;%      [rpm/V]     Electric Motor Speed Constant

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        Imax=EM_NC;
        Vmax=EM_NV;
        Kvmin_N=Kvmax_N;
        Kvmax=Kvmax_N*(2*pi/60);%      [(rad/s)/V]   Max EM speed
constant
        Kvmin=Kvmin_N*(2*pi/60);%      [(rad/s)/V]   Min EM speed
constant
        Qmax=100;%         [N-m]       Max shaft torque
        Qmin=0;%           [N-m]       Min shaft torque
        Imin=I0;%          [A]         Min electric motor current
        %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    if ctrR==length(R_Range)+1 && EM_Case==2
        EM_Case=1;
        Plot_Switch=Initial_Plot_Switch;
    end
    %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Internal Combustion Engine and Cruise
    Pcruise=300;%      [W]         Power Required for cruise
    Vcruise=43;%       [knots]     Cruise Velocity 40-50 knots
    Nmax_ICE=16000;%   [rpm]       Max internal combustion engine rpm
    Nmin_ICE=1500;%    [rpm]       Min internal combustion engine rpm
```

```
    N_ICE=5000;%       [rpm]       Optimum internal combustion engine
Speed

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Clutch Efficiency
    Eff_Clutch=.99;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Regeneration Power Required
    PGen=75;%        [W]         Power Required for generator

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    if Test_Case==2 && ctrR==1
        % Select Altitude for the Calculations
        disp(' ');
        h_TO=input('Enter takeoff altitude (meters AMSL):  ');
        %Note: Bagram Airfield, Afghanistan = 1492m
        %      Kandahar International Airport, Afghanistan = 1017m
        %      Joint Base Balad, Iraq = 49m
        %      Wright-Patterson AFB, OH = 251m
        %      Source: WorldAeroData.com
        h_AGL=input('Enter mission altitude (meters AGL):  ');
        disp(' ');

        h=h_TO + h_AGL;


        [T_TO, a_TO, P_TO, rho_TO] = atmosisa(h_TO);
        [T, a, P, rho] = atmosisa(h);
        disp(['Mission Altitude Density (kg/m^3) = ', num2str(rho)]);
        disp(' ');
    end
    %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %Propeller Data
    if Test_Case==2 && ctrR==1
        disp(' ');
        disp('Select Propeller:');
        disp('   1:  APC    8.8x8.75 in');
        disp('   2:  Rev-Up 12x8     in');
        disp('   3:  APC    12x12    in');
        disp('   4:  APCE   12x12    in');
        disp('   5:  APCE   15x10    in');
        disp('   6:  APCE   16x12    in');
```

```
        disp('    7:  APC    18x8     in');
        disp('    8:  APCE   18x8     in');
        disp('    9:  APC    18x12    in');
        disp('   10:  APCE   18x12    in');
        disp('   11:  APC    20x10    in');
        disp('   12:  APC    20x12    in');

        Prop=input('Enter your selection:  ');
        syms APC APCE ReUp

        if Prop==1
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APC8_8x8_75();
        elseif Prop==2
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=RevUp12x8();
        elseif Prop==3
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APC12x12();
        elseif Prop==4
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APCE12x12();
        elseif Prop==5
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APCE15x10();
        elseif Prop==6
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APCE16x12();
        elseif Prop==7
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APC18x8();
        elseif Prop==8
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APCE18x8();
        elseif Prop==9
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APC18x12();
        elseif Prop==10
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APCE18x12();
        elseif Prop==11
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APC20x10();
        elseif Prop==12
            [Prop_Brand,Prop_Dia,Prop_Pitch,Prop_Data]=APC20x12();
        end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        Prop_D=0.0254*Prop_Dia;%      [m]        Propeller Diameter
        Prop_Rad=Prop_D/2;%           [m]        Properller Radius
        Prop_J_Data=Prop_Data(:,1);%    Advance Ratio   J=V/(n*D)
n=rev/sec
        Prop_Eff_Data=Prop_Data(:,2);%  Prop Efficiency
Eff=J*CT/CP=T*V/P
        Prop_CT_Data=Prop_Data(:,3);%   Thrust Coeff
CT=T/(rho*n^2*D^4)
        Prop_CP_Data=Prop_Data(:,4);%   Power Coeff
CP=P/(rho*n^3*D^5)
        Prop_CQ_Data=Prop_CP_Data/(2*pi);%Torque Coeff
CQ=Q/(rho*n^2*D^5)
    end
    if EM_Case==2 && ctrR==1
    h_wait = waitbar(0,'Please wait...'); % Shows how much time is left
```

```
    end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    wmax_ICE=Nmax_ICE*(2*pi/60);%  [rad/s]       Max ICE speed
    wmin_ICE=Nmin_ICE*(2*pi/60);%  [rad/s]       Min ICE speed
    ICE_w_Range=(wmin_ICE:.01:wmax_ICE);
    ICE_n_Range=ICE_w_Range/(2*pi);
    w_ICE=N_ICE*(2*pi/60);%        [rad/s]       Optimum ICE speed
    n_ICE=N_ICE/60;%               [rps]         Optimum ICE speed

    if ctrR==1
        Vend=Vend*0.5144444;%          [m/s]         Endurance Velocity
    end
    Vcruise=Vcruise*0.5144444;%    [m/s]         Cruise Velocity

    if EM_Case==2
    GR_Case=2;
    GR=1;
    end
    %%%%DO NOT
CHANGE%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    % Assume the prop is 100% efficient in order to determine a
starting
    % minimum voltage(ii=1).  The EM output power will be updated later
    % voltage with the true propeller efficiency to determine the
actual
    % minimum (ii=2).
    Pout_EM=Pend;

    if Test_Case==1
        iter=1;
    elseif Test_Case==2
        iter=2;
    end
    for ii=1:iter
        % calculate the minimum voltage needed to produce the required
        % power at the maximum current
        if iter==1
            Vmin=.1;
        else
            Vmin=Pout_EM/Imax; %        [V]      Min electric motor
voltage
        end
        if ii==1
            if EM_Case==1
                wmin=max((Vmin-Imax*R)*Kvmin*GR,0);%[rad/s]  Max
electric
                                                  %         motor rpm
                wmax=(Vmax-Imin*R)*Kvmax*GR;%       [rad/s]  Min
electric
```

```
            end                                       %         motor rpm
        end

        for ij=1:iter
            % for ij=1, the fmincon optimization routine will determine
the
            % optimal efficiency of an electric motor at the specified
            % parameter ranges.  A propeller will then be added to the
            % electric motor.  The propeller thrust required will be
            % determined based on the specified endurance power and
            % velocity.  From the thrust required and the propeller
            % coefficient data supplied by Wichita State University,
the
            % optimal torque and rotational speed required to produce
the
            % endurace is determined. Torque and rotational speed is
then
            % feed back into the fmincon optimization routine for ij=2
and
            % the optimal electric motor designe, based on the selected
            % propeller is determined.
            %
            % fmincon optimization
            % Set options:
            options=optimset('Display','iter-
detailed','Algorithm','sqp');

            % Run fmincon optimization
            if GR_Case==1
                % starting point
                x0=[wmin;Kvmin;(Vmax-(wmin/Kvmin))/R;GRmin];
                [x_def,FunValue,ExitFlag]=fmincon(@Design_obj,x0,...
                    [],[],[],[],[],[],@Design_const2,options);
            elseif GR_Case==2
                x0=[wmin;Kvmin;(Vmax-(wmin/Kvmin))/R]; % starting point
                [x_def,FunValue,ExitFlag]=fmincon(@Design_obj,x0,...
                    [],[],[],[],[],[],@Design_const1,options);
            end


            ExitFlag

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            % Output Variables from optimization routine
            w_EM=x_def(1);%        [rad/s]     EM Rotation Rate
            Kv=x_def(2);%          [(rad/s)/V] EM Motor Speed Constant
            I=x_def(3);%           [A]         Required Current
            if GR_Case==1
                GR=x_def(4);
            end
            EM_Eff=-FunValue;%     []          EM Efficiency
            EM_Eff_Loss=1-EM_Eff;% []          EM Efficiency Loss
```

```
              % Calculate Shaft Speed, Motor Torque, Shaft Power, Voltage
and
              % Electric Power
              N_EM=w_EM/(2*pi/60);%            [rpm]   EM Rotation Rate
              n_EM=w_EM/(2*pi);%               [rps]   EM Rotation Rate
              w_Shaft_End=w_EM/GR;%            [rad/s] Shaft Rotation
Rate
              N_Shaft_End=w_Shaft_End/(2*pi/60);% [rpm]  Shaft Rotation
Rate
              n_Shaft_End=w_Shaft_End/(2*pi);%    [rps]  Shaft Rotation
Rate
              Kv_N=Kv/(2*pi/60);%              [rpm/V]EM Motor Speed
Constant
              Q_EM=(I-I0)/Kv;%                 [N*m] EM Torque
              Qmax_EM=(Imax-I0)/Kv;%           [N*m] Max Continuous EM
Torque
              Qmax_Shaft=(Imax-I0)/Kv*GR;%     [N*m] Max Contin shaft
Torque
              Qmin_EM=max((Imin-I0)/Kv*GR,0);% [N*m] Min EM Torque
              Q_Shaft_End=Q_EM*GR;%            [N*m] Shaft Torque
              % This is the theoretical max continuous torque.  In
actuality
              % it would ne less due to temperature constraints of the
              % windings.
              if ii==2 && EM_Case==1 && GR_Case==2 && Test_Case==1
                  if abs(Q_Shaft_End-Prop_Q_End)<.01
                      disp('')
                      disp('ERROR!')
                      disp('The torque required to rotate the propeller
is');
                      disp('more than the EM can provide.')
                      disp('Either increase the max current, increase
the')
                      disp(['gear ratio to at least '...
                          ,num2str(Prop_Q_End/Q_Shaft_End*GR),...
                          ', or try the'])
                      disp('option to optimize the gear ratio.')
                      disp('')
                      disp('Press Ctrl C to Exit')
                      disp('ERROR!')
                      disp('')
                      pause
                  end
              end

              V=w_EM/Kv+I*R;%                  [Volts] Required Voltage
              Vemf=w_EM/Kv;%                   [V]     Back-emf
                          %                            (electromotive
force)
              Q_Stall=(V-I0*R)/(Kv*R);%        [N*m]   Stall Torque
              w0=V*Kv;%                        [rad/s] No-Load Speed
              I_Stall=(Q_Stall*Kv)+I0;%        [A]     Staring Current
              I_maxEff=(I0*I_Stall)^.5;%       [A]     Most Efficient
Current
                                  %                    at V
```

```
            w_maxEff=w0-I_maxEff*R*Kv;%      [rad/s] Most Efficient
Speed
                                        %            at V
            N_maxEff=w_maxEff/(2*pi/60);%    [rpm]   Most Efficient
Speed
                                          %          at V
            Effmax_EM=((I_maxEff-I0)/I_maxEff)^2;%   EM Max Efficiency
            N0=w0/(2*pi/60);%                [rpm]   No-Load Speed
            Pin_EM=V*I;%                     [W]     Electric Power
Input
            Pout_EM=w_EM*Q_EM;%              [W]     Shaft Power
            wmin_EM=max((Vmin-Imax*R)*Kv*GR,0);
            wmax_EM=(Vmax-Imin*R)*Kv*GR;
            % Calculate Generator Speed Range
            wmax_GEN=wmax_ICE*GR;%         [rad/s]  Max Gen rotation
rate
            wmin_GEN=wmin_ICE*GR;%         [rad/s]  Min Gen rotation
rate
            Nmax_GEN=Nmax_ICE*GR;%         [rpm]    Max Gen rotation
rate
            Nmin_GEN=Nmin_ICE*GR;%         [rpm]    Min Gen rotation
rate
            if ii==2 && EM_Case==1
            if abs(w_Shaft_End-w_Prop_End)>.00001
               disp('')
               disp('ERROR!')
               disp('The required rotation rate for endurance exceeds
');
               disp('the max rotation rate of the EM.')
               disp('Either increase the max voltage, choose a larger
')
               disp('propeller, or increase the gear ratio.')
               disp('')
               disp('Press Ctrl C to Exit')
               disp('ERROR!')
               disp('')
               disp('ERROR!')
               disp('')
               pause
             end
             end

            C_Total=C_bat*Num_bat;%          [mA-h]  Total Battery
Capacity
            tend=C_Total/1000/I*3600;%       [sec]   Endurance time

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
            %%
            if Plot_Switch==1;
                % a) Plot w and I for a const Kv with units of rad/s
                if Test_Case==1 || Test_Case==2 && ii==1 && ij==1 ||...
                        Test_Case==2 && ii==2 && ij==2
                if ii==1 && ij==1
                    figa=1;
                elseif ii==2 && ij==2
```

```
        figa=4;
    else figa=5000;
    end
    %         end

    if figa==1 || figa==4
        figure(figa)
        % Create Grid
        axmin=wmin_EM-(wmax_EM/5);
        axmax=wmax_EM+(wmax_EM/5);
        aymin=0-Imax/5;
        aymax=Imax+Imax/5;

        [wa,Ia]=meshgrid(axmin:axmax,aymin:aymax);
        % Enter Cost and Constraint Functions
        % Maximize the efficiency
        fa=(((Ia-
I0).*(wa./Kv))./(((wa./Kv)+(Ia.*R)).*Ia))*100;
        g2a=-wa+wmin;% min rotation speed
        if wmin~=wmax
            g2a=-wa+0;% min rotation speed
            if EM_Case==1
                g1a=wa-wmax;% max rotation speed
            end
            g3a=Ia-Imax;% max current
            g4a=-Ia+Imin;% min current
            g7a=(Ia-I0)./Kv-Qmax;% max torque
            g8a=-((Ia-I0)./Kv)+Qmin;% min torque
        end
        g5a=(wa./Kv+Ia.*R)-Vmax;% max voltage
        g6a=-(wa./Kv+Ia.*R)+Vmin;% min voltage

        if wmin==wmax
            g9a=Ia-I;% current
            g10a=(Ia-I0)./Kv-Qmax_EM;% max torque
            g11a=-((Ia-I0)./Kv)+Qmin_EM;% min torque
            if EM_Case==1
                g12a=wa-wmax_EM;% max rotation speed
            end
            g13a=-wa+wmin_EM;% min rotation speed
        end

        % Prepare Plot
        xlabel('Rotational Speed, \omega [rad/s]')
        ylabel('Current, I [Amps]')
        if wmin~=wmax
            title({'Electric Motor Efficiency',...
   'Rotational Speed vs Current for Fixed Motor Speed
Constant'})
        else
            title({'Electric Motor Efficiency',...
   'Rotational Speed vs Current for Fixed Motor Speed
Constant'})
        end
```

```
grid on; hold on
% Plot Cost and Constraint Functions
% cv:  contour values
cvfa=[0 10 20 30 40 50 60 70 80 85 86 87 88 89 90 ...
    91 92 93 94 95 100 -FunValue*100];
costa=contour(wa,Ia,fa,cvfa,'k');
clabel(costa)
% plot min rotational speed
const2a1=contour(wa,Ia,g2a,[0 0],'r','LineWidth',2.5);
if wmin~=wmax
    if EM_Case==1
        % plot max rotation speed
        const1a1=contour(wa,Ia,g1a,[0 0],'r',...
            'LineWidth',2.5);
    end
    % plot max current
    const3a1=contour(wa,Ia,g3a,[0 0],'g',...
        'LineWidth',2.5);
    % plot min current
    const4a1=contour(wa,Ia,g4a,[0 0],'g',...
        'LineWidth',2.5);
    % plot max torque
    const7a1=contour(wa,Ia,g7a,[0 0],'y',...
        'LineWidth',2.5);
    % plot min torque
    const8a1=contour(wa,Ia,g8a,[0 0],'y',...
        'LineWidth',2.5);
end
% plot max voltage
const5a1=contour(wa,Ia,g5a,[0 0],'m',...
    'LineWidth',2.5);
% plot min voltage
const6a1=contour(wa,Ia,g6a,[0 0],'m',...
    'LineWidth',2.5);

if wmin==wmax
    % plot current
    const9a1=contour(wa,Ia,g9a,[0 0],'g',...
        'LineWidth',2.5);
    % plot max torque
    const10a1=contour(wa,Ia,g10a,[0 0],'y',...
        'LineWidth',2.5);
    % plot min torque
    const11a1=contour(wa,Ia,g11a,[0 0],'y',...
        'LineWidth',2.5);
    if EM_Case==1
        % plot max rotation speed
        const12a1=contour(wa,Ia,g12a,[0 0],'r',...
            'LineWidth',2.5);
    end
    %plot min rotation speed
    const13a1=contour(wa,Ia,g13a,[0 0],'r',...
```

```
                    'LineWidth',2.5);
            end

            legend('Efficiency')

            if wmin==wmax
                if EM_Case==1
                    text(wmax_EM,aymin+(.5*(Imin-aymin)),...
                        'Max Rotational Speed',...
                        'HorizontalAlignment','center',...

'BackgroundColor','w','EdgeColor','r',...
                        'LineWidth',2.5);
                end
                text(wmin_EM,aymin+(.5*(Imin-aymin)),...
                    'Min Rotational Speed',...
                    'HorizontalAlignment','center',...
                    'BackgroundColor','w','EdgeColor','r',...
                    'LineWidth',2.5);
                text(w_EM,aymin+(.5*(Imin-aymin)),...
                    'Rotational
Speed','HorizontalAlignment',...

'center','BackgroundColor','w','EdgeColor',...
                    'r','LineWidth',2.5);
                text(.5*(axmax-w_EM)+w_EM,Kv*Q_EM+I0,...
                    ' Torque
','HorizontalAlignment','center',...
                    'BackgroundColor','w','EdgeColor','y',...
                    'LineWidth',2.5);
                text(.5*(axmax-w_EM)+w_EM,Kv*Qmax_EM+I0,...
                    ' Max Torque ','HorizontalAlignment',...

'center','BackgroundColor','w','EdgeColor',...
                    'y','LineWidth',2.5);
                text(.5*(axmax-w_EM)+w_EM,Kv*Qmin_EM+I0,...
                    ' Min Torque ','HorizontalAlignment',...

'center','BackgroundColor','w','EdgeColor',...
                    'y','LineWidth',2.5);
                text(.5*(wmin_EM-axmin)+axmin,I,' Current ',...
                    'HorizontalAlignment','center',...
                    'BackgroundColor','w','EdgeColor','g',...
                    'LineWidth',2.5);
            else
                if EM_Case==1
                    text(wmax,aymin+(.5*(Imin-aymin)),...
                        'Max Rotational Speed',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor',...
                        'r','LineWidth',2.5);
                end
                text(wmin,aymin+(.5*(Imin-aymin)),...
                    'Min Rotational Speed',...
                    'HorizontalAlignment','center',...
```

```
                            'BackgroundColor','w','EdgeColor','r',...
                            'LineWidth',2.5);
                        text(axmax+(.5*(wmax_EM-axmax)),Kv*Qmax+I0,...
                            ' Max Torque ','HorizontalAlignment',...

'center','BackgroundColor','w','EdgeColor',...
                            'y','LineWidth',2.5);
                        text(axmax+(.5*(wmax_EM-axmax)),Kv*Qmin+I0,...
                            ' Min Torque ','HorizontalAlignment',...

'center','BackgroundColor','w','EdgeColor',...
                            'y','LineWidth',2.5);
                    end
                    text(.5*(wmin_EM-axmin)+axmin,Imax,...
                        ' Max Current
','HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','g',...
                        'LineWidth',2.5);
                    text(.5*(wmin_EM-axmin)+axmin,Imin,' Min Current
',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','g',...
                        'LineWidth',2.5);
                    text((Vmax-(.75*(Imax-Imin)+Imin)*R)*Kv,.75*...
                        (Imax-Imin)+Imin,' Max Voltage ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','m',...
                        'LineWidth',2.5);
                    text((Vmin-(.25*(Imax-Imin)+Imin)*R)*Kv,.25*...
                        (Imax-Imin)+Imin,' Min Voltage ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','m',...
                        'LineWidth',2.5);

                    text(w_EM,I,'X','FontSize',24,'FontWeight',...
                        'bold','HorizontalAlignment','center')
                    hold off
                end
                end
                %%
                % b) Plot w and Kv for a const I with units of rad/s
                if Test_Case==1 || Test_Case==2 && ii==1 && ij==1 ||...
                        Test_Case==2 && ii==2 && ij==2
                if ii==1 && ij==1
                    figb=2;
                elseif ii==2 && ij==2
                    figb=5;
                else figb=5000;
                end
                %         end
                if figb==2 || figb==5
                    figure(figb)
                    % Create Grid
                    bxmin=wmin_EM-(wmax_EM/5);
                    bxmax=wmax_EM+(wmax_EM/5);
```

```
                    bymin=Kv-(Kv/5);
                    bymax=Kv+(Kv/5);

                    [wb,Kvb]=meshgrid(bxmin:bxmax,bymin:bymax);
                    % Enter Cost and Constraint Functions
                    % Maximize the efficiency
                    fb=(((I-
I0).*(wb./Kvb))./(((wb./Kvb)+(I.*R)).*I))*100;
                    if EM_Case==1
                        g1b=wb-wmax;% max rotation speed
                    end
                    g2b=-wb+wmin;% min rotation speed
                    g3b=Kvb-Kvmax;% max motor speed constant
                    g4b=-Kvb+Kvmin;% min motor speed constant
                    g5b=(wb./Kvb+I.*R)-Vmax;% max voltage
                    g6b=-(wb./Kvb+I.*R)+Vmin;% min voltage
                    if wmin~=wmax
                        g7b=(I-I0)./Kvb-Qmax;% max torque
                        g8b=-((I-I0)./Kvb)+Qmin;% min torque
                    end
                    if wmin==wmax
                        g9b=wb-(Vmin-I*R)*Kv*GR;% min rotational speed
                        if EM_Case==1
                            g10b=-wb+(Vmax-I*R)*Kv*GR;% max rotation
speed
                        end
                    end
                    g12b=Kvb-Kv;% motor speed constant
                    % Prepare Plot
                    xlabel('Rotational Speed, \omega [rad/s]')
                    ylabel('Motor Speed Constant, Kv [(rad/s)/V]')
                    if wmin~=wmax
                        title({'Electric Motor Efficiency',...
             'Rotational Speed vs Motor Speed Constant for Fixed
Current'})
                    else
                        title({'Electric Motor Efficiency',...
             'Rotational Speed vs Motor Speed Constant for Fixed
Current'})
                    end
                    grid on; hold on
                    % Plot Cost and Constraint Functions
                    % cv:  contour values
                    cvfb=[0 10 20 30 40 50 60 70 80 85 86 87 88 89 90
...
                        91 92 93 94 95 100 -FunValue*100];
                    costb=contour(wb,Kvb,fb,cvfb,'k');
                    clabel(costb)
                    if EM_Case==1
                        % plot max rotation speed
                        const1b1=contour(wb,Kvb,g1b,[0 0],'r',...
                            'LineWidth',2.5);
                    end
                    % plot min rotation speed
```

```
                    const2b1=contour(wb,Kvb,g2b,[0
0],'r','LineWidth',2.5);
                    % plot max motor speed constant
                    const3b1=contour(wb,Kvb,g3b,[0
0],'b','LineWidth',2.5);
                    % plot min motor speed constant
                    const4b1=contour(wb,Kvb,g4b,[0
0],'b','LineWidth',2.5);
                    % plot max voltage
                    const5b1=contour(wb,Kvb,g5b,[0
0],'m','LineWidth',2.5);
                    % plot min voltage
                    const6b1=contour(wb,Kvb,g6b,[0
0],'m','LineWidth',2.5);

                    if wmin~=wmax
                        % plot max torque
                        const7b1=contour(wb,Kvb,g7b,[0 0],'y',...
                            'LineWidth',2.5);
                        % plot min torque
                        const8b1=contour(wb,Kvb,g8b,[0 0],'y',...
                            'LineWidth',2.5);
                    end
                    if wmin==wmax
                        % plot min rotation speed
                        const9b1=contour(wb,Kvb,g9b,[0 0],'r',...
                            'LineWidth',2.5);
                        if EM_Case==1
                            % plot max rotation speed
                            const10b1=contour(wb,Kvb,g10b,[0 0],'r',...
                                'LineWidth',2.5);
                        end
                    end
                    % plot motor speed constant
                    const12b1=contour(wb,Kvb,g12b,[0 0],'b',...
                        'LineWidth',2.5);
                    legend('Efficiency [%]')

                    if wmin==wmax
                        if EM_Case==1
                            text((Vmax-I*R)*Kv*GR,bymin+(.5*...
                                (Kvmin-bymin)),' Max Rotational Speed
',...
                                'HorizontalAlignment','center',...
                                'BackgroundColor','w','EdgeColor',...
                                'r','LineWidth',2.5);
                        end
                        text((Vmin-I*R)*Kv*GR,bymin+(.5*(Kv-bymin)),...
                            ' Min Rotational Speed ',...
                            'HorizontalAlignment','center',...
                            'BackgroundColor','w','EdgeColor','r',...
                            'LineWidth',2.5);
                        text(w_EM,bymin+(.5*(Kv-bymin)),...
                            ' Rotational Speed
','HorizontalAlignment',...
```

```
'center','BackgroundColor','w','EdgeColor',...
                        'r','LineWidth',2.5);
                    text(.5*(wmax_EM-wmin_EM)+wmin_EM,Kv,' Kv ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','b',...
                        'LineWidth',2.5);
                    text((Kv+.5*(bymax-Kv))*(Vmax-I*R),(Kv+.5*...
                        (bymax-Kv)),'Max Voltage',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','m',...
                        'LineWidth',2.5);
                    text((Kv+.5*(bymax-Kv))*(Vmin-I*R),(Kv+.5*...
                        (bymax-Kv)),'Min Voltage',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','m',...
                        'LineWidth',2.5);
                else
                    if EM_Case==1
                        text(wmax,bymin+(.5*(Kvmin-bymin)),...
                            ' Max Rotational Speed ',...
                            'HorizontalAlignment','center',...

'BackgroundColor','w','EdgeColor','r',...
                            'LineWidth',2.5);
                    end
                    text(wmin,bymin+(.5*(Kv-bymin)),...
                        ' Min Rotational Speed ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','r',...
                        'LineWidth',2.5);
                    text(.5*(wmin_EM-bxmin)+bxmin,Kv,' Kv ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','b',...
                        'LineWidth',2.5);
                    text((bymin+(.5*(Kvmin-bymin)))*(Vmax-I*R),...
                        bymin+(.5*(Kvmin-bymin)),'Max Voltage',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','m',...
                        'LineWidth',2.5);
                    text((bymax-(.5*(Kvmin-bymin)))*(Vmin-I*R),...
                        bymax-(.5*(Kvmin-bymin)),'Min Voltage',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','m',...
                        'LineWidth',2.5);
                    text(bxmax+(.5*(wmax_EM-bxmax)),(I-I0)/Qmax,...
                        'Max
Torque','HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','y',...
                        'LineWidth',2.5);
                    text(bxmax+(.5*(wmax_EM-bxmax)),(I-I0)/Qmin,...
                        'Min
Torque','HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','y',...
                        'LineWidth',2.5);
```

```
        end

        text(w_EM,Kv,'X','FontSize',24,'FontWeight',...
            'bold','HorizontalAlignment','center')
        hold off
    end
    end

    %%
    % c) Plot I and Kv for a const w with units of rad/s
    if Test_Case==1 || Test_Case==2 && ii==1 && ij==1 ||...
            Test_Case==2 && ii==2 && ij==2
    if ii==1 && ij==1
        figc=3;
    elseif ii==2 && ij==2
        figc=6;
    else figc=5000;
    end
    %         end
    if figc==3 || figc==6
        figure(figc)
        cxmin=0-Imax/5;
        cxmax=Imax+(Imax/5);
        cymin=Kv-(Kv/5);
        cymax=Kv+(Kv/5);

        [Ic,Kvc]=meshgrid(cxmin:cxmax,cymin:cymax);
        % Enter Cost and Constraint Functions
        % Maximize the efficiency
        fc=(((Ic-I0).*(w_EM./Kvc))./(((w_EM./Kvc)+...
            (Ic.*R)).*Ic))*100;
        g1c=Ic-Imax;% max current
        g2c=-Ic+Imin;% min current
        g3c=Kvc-Kvmax;% max motor speed constant
        g4c=-Kvc+Kvmin;% min motor speed constant
        g5c=(w_EM./Kvc+Ic.*R)-Vmax;% max voltage
        g6c=-(w_EM./Kvc+Ic.*R)+Vmin;% min voltage
        g7c=(Ic-I0)./Kvc-Qmax;% max torque
        g8c=-((Ic-I0)./Kvc)+Qmin;% min torque
        if wmin==wmax
            g9c=(Ic-I0)./Kvc-Qmax_EM;% max torque
            g10c=-((Ic-I0)./Kvc)+Qmin_EM;% min torque
            g11c=Ic-I;% current
        end
        g12c=Kvc-Kv;% motor speed constant
        % Prepare Plot
        xlabel('Current, I [Amps]')
        ylabel('Motor Speed Constant, Kv [(rad/s)/V]')
        if wmin~=wmax
            title({'Electric Motor Efficiency',...
    'Current vs Motor Speed Constant for Fixed Rotational
Speed'})
        else
            title({'Electric Motor Efficiency',...
```

```
            'Current vs Motor Speed Constant for Fixed Rotational
Speed'})
                    end
                    grid on; hold on
                    % Plot Cost and Constraint Functions
                    % cv:  contour values
                    cvfc=[0 10 20 30 40 50 60 70 80 85 86 87 88 89 90
...
                        91 92 93 94 95 100 -FunValue*100];
                    costc=contour(Ic,Kvc,fc,cvfc,'k');
                    clabel(costc)
                    % plot max current
                    const1c1=contour(Ic,Kvc,g1c,[0
0],'g','LineWidth',2.5);
                    % plot min current
                    const2c1=contour(Ic,Kvc,g2c,[0
0],'g','LineWidth',2.5);
                    % plot max motor speed constant
                    const3c1=contour(Ic,Kvc,g3c,[0
0],'b','LineWidth',2.5);
                    % plot min motor speed constant
                    const4c1=contour(Ic,Kvc,g4c,[0
0],'b','LineWidth',2.5);
                    % plot max voltage
                    const5c1=contour(Ic,Kvc,g5c,[0
0],'m','LineWidth',2.5);
                    % plot min voltage
                    const6c1=contour(Ic,Kvc,g6c,[0
0],'m','LineWidth',2.5);
                    % plot max torque
                    const7c1=contour(Ic,Kvc,g7c,[0
0],'y','LineWidth',2.5);
                    % plot min torque
                    const8c1=contour(Ic,Kvc,g8c,[0
0],'y','LineWidth',2.5);

                    if wmin==wmax
                        % plot max torque
                        const9c1=contour(Ic,Kvc,g9c,[0 0],'y',...
                            'LineWidth',2.5);
                        % plot min torque
                        const10c1=contour(Ic,Kvc,g10c,[0 0],'y',...
                            'LineWidth',2.5);
                        % plot current
                        const11c1=contour(Ic,Kvc,g11c,[0 0],'g',...
                            'LineWidth',2.5);
                    end
                    % plot motor speed constant
                    const12c1=contour(Ic,Kvc,g12c,[0 0],'b',...
                        'LineWidth',2.5);
                    legend('Efficiency [%]');
                    if wmin==wmax
                        text((Kv+.5*(cymax-Kv))*Q_EM+I0,...
                            (Kv+.5*(cymax-Kv)),...
```

```
                            ' Torque
','HorizontalAlignment','center',...
                            'BackgroundColor','w','EdgeColor','y',...
                            'LineWidth',2.5);
                        text((.5*(cymax-Kv)+Kv)*Qmax_EM+I0,.5*...
                            (cymax-Kv)+Kv,' Max Torque ',...
                            'HorizontalAlignment','center',...
                            'BackgroundColor','w','EdgeColor','y',...
                            'LineWidth',2.5);
                        text((.5*(cymax-Kv)+Kv)*Qmin_EM+I0,.5*...
                            (cymax-Kv)+Kv,' Min Torque ',...
                            'HorizontalAlignment','center',...
                            'BackgroundColor','w','EdgeColor',...
                            'y','LineWidth',2.5);
                        text(I,.5*(Kv-cymin)+cymin,' Current ',...
                            'HorizontalAlignment','center',...
                            'BackgroundColor','w','EdgeColor','g',...
                            'LineWidth',2.5);
                    else
                        text(.5*(cxmax-I)+I,((.5*(cxmax-I)+I)-
I0)/Qmax,...
                            ' Max Torque ','HorizontalAlignment',...

'center','BackgroundColor','w','EdgeColor',...
                            'y','LineWidth',2.5);
                        text(.5*(cxmax-I)+I,((.5*(cxmax-I)+I)-
I0)/Qmin,...
                            ' Min Torque ','HorizontalAlignment',...

'center','BackgroundColor','w','EdgeColor',...
                            'y','LineWidth',2.5);
                    end
                    text(cxmin+(.5*(Imin-cxmin)),Kv,' Kv ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','b',...
                        'LineWidth',2.5);
                    text(Imax,.5*(Kv-cymin)+cymin,' Max Current ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','g',...
                        'LineWidth',2.5);
                    text(Imin,.5*(Kv-cymin)+cymin,' Min Current ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','g',...
                        'LineWidth',2.5);

                    text((Vmax-w_EM/(cymin+(.75*(Kv-cymin))))/R,...
                        cymin+(.75*(Kv-cymin)),' Max Voltage ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','m',...
                        'LineWidth',2.5);
                    text((Vmin-w_EM/(cymin+(.5*(Kv-cymin))))/R,...
                        cymin+(.5*(Kv-cymin)),' Min Voltage ',...
                        'HorizontalAlignment','center',...
                        'BackgroundColor','w','EdgeColor','m',...
                        'LineWidth',2.5);
```

```
            text(I,Kv,'X','FontSize',24,'FontWeight','bold',...
                'HorizontalAlignment','center')
            hold off
        end
        end
    end

    %%
    % Plot Motor Torque Vs Motor Speed for various amounts of
    % applied voltage
    if ii==1 && ij==1
        Vmax2=Vmax;
    end
    if Test_Case==2 && ii==1 && ij==1
        Vmax=100;
    end

    EM_V_Range=sort([5:5:Vmax-1 V Vmax]);%         [V]
voltage
    if length(EM_V_Range)>10
        EM_V_Range=sort([10:10:Vmax-1 V Vmax]);%   [V]
voltage
    end

    if iter==1 && Test_Case==1
        syms yes;
        syms Yes;
        syms no;
        syms No;
        NVR=input...
            ('Do you want to enter a specific voltage range?:
');

        if NVR==yes || NVR==Yes
            disp('Sample Voltage Range')
            disp('[1 3 10 15 20]')
            disp('[1:20]')
            disp('[1 5:5:20]')
            disp('')
            EM_V_Range=input...
                ('Enter Voltage Matrix:  ');%  [V]     voltage
            EM_V_Range=sort(EM_V_Range);
            EM_V_Range=EM_V_Range(EM_V_Range<=Vmax);

        elseif NVR==no || NVR==No
%                 EM_V_Range=EM_V_Range;
        end
    end

    if EM_V_Range(end)-EM_V_Range(end-1)<1
        EM_V_Range=EM_V_Range(:,1:end-1);
    end

    EM_I_Range=[I0:Imax/20:Imax Imax];%        [A]     Current
```

```
            EM_Q_Range=(EM_I_Range-I0)/Kv; %           [N*m]   Torque
Range
            EM_P_Range=min(EM_I_Range)*min(EM_V_Range)...
                :max(EM_I_Range)*max(EM_V_Range);%     [W]     Power
Range
            EM_w_Range=Kv*((R*I0:.001:Vmax)-R*I0);%    [rad/s] EM No-
Load
                                                %              Speed
Range
            EM_N_Range=EM_w_Range/(2*pi/60);% [rps] EM No-Load speed
range
            EM_n_Range=EM_w_Range/(2*pi);%    [rpm] EM No-Load speed
range

            % equations form Drela
            clear EM_Current_vs_volt EM_Torque_vs_volt EM_Power_vs_volt
            clear EM_Eff_vs_volt EEE EM_T_V EM_E_V
            for ctr=1:length(EM_V_Range)
                EM_Current_vs_volt(:,ctr)=(EM_V_Range(ctr)-...
                    (EM_w_Range./Kv))/R;
                EM_Torque_vs_volt(:,ctr)=((EM_V_Range(ctr)-...
                    (EM_w_Range./Kv))/R-I0)/Kv;
                EM_Power_vs_volt(:,ctr)=((EM_V_Range(ctr)-...
                    (EM_w_Range./Kv))/R-I0).*EM_w_Range./Kv;
                EM_Eff_vs_volt(:,ctr)=(1-(I0*R./(EM_V_Range(ctr)-...

(EM_w_Range./Kv)))).*(EM_w_Range/(EM_V_Range(ctr)*Kv));
            end
            if ij==2
                Vmax=Vmax2;
            end
            EM_Power_vs_volt(EM_Power_vs_volt<0)=0;
            EM_Current_vs_volt(EM_Current_vs_volt<0)=0;
            EM_Torque_vs_volt(EM_Torque_vs_volt<0)=0;
            EM_Eff_vs_volt(EM_Eff_vs_volt<0)=0;
            EM_Eff_vs_volt(EM_Eff_vs_volt>1)=0;

            % Calculate the efficiency map
            EM_w_Range_small=[1:10 0:10:wmax_EM];
            [EMQR,EMwR]=meshgrid(EM_Q_Range,EM_w_Range_small);
            [EMIR,EMwR]=meshgrid(EM_I_Range,EM_w_Range_small);
            Motor_outpwr_map=EMQR.*EMwR;
            Motor_inpwr_map=EMIR.*((EMwR/Kv)+R*EMIR);
            Motor_Eff_Map=Motor_outpwr_map./Motor_inpwr_map;
            %%
            if Plot_Switch==1;
                if Test_Case==1 || Test_Case==2 && ii~=1 && ij~=1
                figure(7)
                C=mesh(EMwR,EMQR,Motor_Eff_Map);
                set(gca,'GridLineStyle','--');
                xlabel('\omega [rad/s]')
                ylabel('Torque [N\cdotm]')
                zlabel('Efficiency');
                title('Motor Efficiency Map');
```

```
        end
    end
    %%
    if Plot_Switch==1;
        if Test_Case==1 || Test_Case==2 && ii~=1 && ij~=1
        figure(8)
        C=contour(EMwR,EMQR,Motor_outpwr_map,20);
%         hold on
%         plot(w_EM,Q_EM,'X')
        xlabel('Rotational Speed, \omega [rad/s]')
        ylabel('Torque (N\cdotm)');
        legend('Motor Power, P_S_h_a_f_t [W]')
        clabel(C);
        grid on
        end
    end
    %%
    if Plot_Switch==1;
         if Test_Case==1 || Test_Case==2 && ii~=1 && ij~=1
        figure(9);
        C=contour(EMwR,EMQR,Motor_Eff_Map*100,...
            [10 20 30 40 50 60 70 80 85 86 87 88 89 90 91 92 93
...

            94 95 100 EM_Eff*100]);
        %         hold on
        %         plot(w_EM,Q_EM,'X')
        xlabel('Rotational Speed, \omega [rad/s]')
        ylabel('Torque (N\cdotm)')
        zlabel('Efficiency (%)');
        clabel(C);
        grid on
        legend('Efficiency');
         end
    end



    %%

    if Plot_Switch==1;
        if Test_Case==1 || Test_Case==2 && ii~=1 && ij~=1

        for kk=1:2
            % Plot Motor Torque Vs Motor Speed for various
amounts

            % of applied voltage
            if kk==1
                fig=10;
            end
            if kk==2
                fig=11;
            end
            figureX = figure(fig);
            % Create axes 1
```

```
xlimax1=[0 max(EM_w_Range)];
if kk==1
    ylimax1=([min(min(EM_Torque_vs_volt))
        max(max(EM_Torque_vs_volt))]);
else
    ylimax1=([min(EM_Q_Range) max(EM_Q_Range)]);
end
ytickax1=(min(ylimax1):(max(ylimax1)-...
    min(ylimax1))/3:max(ylimax1));
ax1 = axes('Parent',figureX,...
    'Position',[0.13 0.7 0.42 0.2],'XColor','k',...
    'YColor','k','ylim',ylimax1,'xlim',xlimax1,...
    'ytick',ytickax1,'yticklabel',...
    sprintf('%.2f |',ytickax1'));

% Plot torque vs speed
line(EM_w_Range,EM_Torque_vs_volt,'Parent',ax1);

ylabel({'Motor Torque,','Q_m [N-m]'});
grid on

ylim1 = get(ax1,'YLim');

ytickax1=get(ax1,'ytick');
yticklabelax1=get(ax1,'yticklabel');

ylimax1_2=ylim1*Kv+I0;
ytickax1_2=ytickax1*Kv+I0;

ax1_2 = axes('Position',get(ax1,'Position'),...
    'XAxisLocation','bottom',...
    'YAxisLocation','right',...
    'Color','none',...
    'XColor','k','YColor','k','ylim',ylimax1_2,...

'xtick',[],'xticklabel',[],'ytick',ytickax1_2,...
    'yticklabel',sprintf('%.2f |',ytickax1_2'));
ylabel({'Motor Current,','I [A]'});
% xlabel('Rotational Speed, \omega [rad/s]');
% Create axes 2
axes2 = axes('Parent',figureX,...
    'Position',[0.13 0.4 0.42 0.2]);
if kk==1
    ylim([0 max(max(EM_Power_vs_volt))])
else
    ylim([0 max(EM_P_Range)])
end
box(axes2,'on');
grid(axes2,'on');
hold(axes2,'all');

% Plot Power vs speed
plot(EM_w_Range,EM_Power_vs_volt,'Parent',axes2);
xlim([0 max(EM_w_Range)])
```

```
            % xlabel('Rotational Speed, \omega [rad/s]');
            ylabel({'Shaft Power,','P_S_h_a_f_t [W]'});

            % Create axes 3
            axes3 = axes('Parent',figureX,...
                'Position',[0.13 0.1 0.42 0.2]);
            xlim([0 max(EM_w_Range)])
            ylim([0 100])
            box(axes3,'on');
            grid(axes3,'on');
            hold(axes3,'all');

            % Plot Efficiency vs speed
            plot(EM_w_Range,EM_Eff_vs_volt*100,'Parent',axes3);

            xlabel('Rotational Speed, \omega [rad/s]');
            ylabel({'Efficiency,',' \eta_m [%]'});

            % create the legned
            leg=num2str((EM_V_Range).');
            [rleg cleg]=size(leg);
            legend1=legend([leg repmat(' Volts',[rleg 1])]);

            set(legend1,'Position'...
                ,[0.5 0.1 0.2 0.3]);
            axes(ax1)
        end
        end
end

%%
if Plot_Switch==1;
    if Test_Case==1 || Test_Case==2 && ii~=1 && ij~=1

    for LL=1:2
        if LL==1
            fig=12;
            EM_I_Range_Large=I0:.1:I_Stall;
        end
        if LL==2
            fig=13;
            EM_I_Range_Large=I0:.1:Imax;
        end

        EM_Q_Range_V=(EM_I_Range_Large-I0)/Kv;
        EM_w_Range_V=(V-EM_I_Range_Large*R)*Kv;
        EM_P_Range_V=EM_w_Range_V.*EM_Q_Range_V;
        EM_Eff_Range_V=EM_P_Range_V./(EM_I_Range_Large*V);

        X_Data=EM_Q_Range_V;
        Y_Data1=EM_w_Range_V;
        Y_Data2=EM_P_Range_V;
        Y_Data3=EM_Eff_Range_V;
```

```
                    figure(fig)
                    % Plot First X-Y
                    H1= plot(X_Data,Y_Data1,'LineStyle','-
','Color','k');
                    L1=get(H1,'LineStyle');
                    C1=get(H1,'Color');
                    hold on
                    grid on

                    % Set First Axes
                    ax(1)=gca;
                    pos1 = [0.1  0.1  0.60 0.8];
                    % rotational speed axis

set(ax(1),'XColor','k','YColor','k','position',pos1)
                    xlim1 = get(ax(1),'XLim');


                    xtickax1=get(ax(1),'xtick');
                    xticklabelax1=get(ax(1),'xticklabel');
                    xlabel('Motor Torque,Q_m [N-m]');
%                     xlabel('Motor Current, I [A]');

                    ylim1 = get(ax(1),'YLim');
                    ax1hv = get(ax(1),'HandleVisibility');

                    xlimax1_2=xlim1*Kv+I0;
                    xtickax1_2=xtickax1*Kv+I0;

%                     xlimax1_2=(xlim1-I0)/Kv;
%                     xtickax1_2=(xtickax1-I0)/Kv;

                    ax1_2 = axes('Position',get(ax(1),'Position'),...
                        'XAxisLocation','top',...
                        'YAxisLocation','right',...
                        'Color','none',...
                        'XColor','k','YColor','k','xlim',xlimax1_2,...

'ytick',[],'yticklabel',[],'xtick',xtickax1_2,...
                        'xticklabel',sprintf('%.2f |',xtickax1_2'));

                        xlabel('Motor Current, I [A]');
%                     xlabel('Motor Torque,Q_m [N-m]');

                    title(['Input Voltage = ',num2str(V),' Volts'])

                    ax(2) =
axes('HandleVisibility',ax1hv,'Position',...
                        get(ax(1),'Position'),'Parent',get(ax(1),...
                        'Parent'));

                    % Plot Second X-Y
```

```
                    H2= plot(X_Data,Y_Data2,'LineStyle','--
','Color','b');
                    L2=get(H2,'LineStyle');
                    C2=get(H2,'Color');

                    %Determine  y-limits for the second axes
                    ytickax1=get(ax(1),'ytick');
                    Ydivision=length(ytickax1)-1;

                    ax2MaxY=ceil(max(Y_Data2));
                    ax2minY=floor(min(Y_Data2));
                    ax2Ydivision=(ax2MaxY-ax2minY)/Ydivision;

                    ylim2=[ax2minY ax2MaxY];
                    ytick2=(ax2minY:ax2Ydivision:ax2MaxY);

                    set(ax(2),'YAxisLocation','right','Color','none',
...
                        'XGrid','off','YGrid','off','Box','off', ...
                        'HitTest','off','YColor','b','XColor','k',...

'xtick',[],'xlim',xlim1,'ylim',ylim2,'ytick',...

ytick2,'xticklabel',xticklabelax1,'yticklabel',...
                        sprintf('%.2f |',ytick2'));

                    % Plot 3=Third X-Y
                    % Set positon of third axes
                    pos3=[0.1  0.1  0.75 0.8];

                    % Determine the scale factor of third axes to first
                    scalefactor=pos3(3)/pos1(3);

                    %Determine x-limits for the third axes
                    xtickax1=get(ax(1),'xtick');
                    xlim3=[xlim1(1) xlim1(2)*scalefactor];

                    %Determine y-limits for the third axes
                    ytickax1=get(ax(1),'ytick');
                    Ydivision=length(ytickax1)-1;

                    ax3MaxY=ceil(max(Y_Data3));
                    ax3minY=floor(min(Y_Data3));
                    ax3Ydivision=(ax3MaxY-ax3minY)/Ydivision;

                    ylim3=[ax3minY ax3MaxY];
                    ytick3=(ax3minY:ax3Ydivision:ax3MaxY);

                    %Set third axes
                    ax(3)=axes('Position',pos3,'box','off',...
                        'Color','none','XColor','k','YColor','r',...

'xtick',xtickax1,'xlim',xlim3,'yaxislocation',...
```

```
                        'right','ylim',ylim3,'ytick',ytick3,...
                        'yticklabel',sprintf('%.2f |',ytick3'));

                    % Plot third X-Y
                    H3 = line(X_Data,Y_Data3,'Color','r','Parent',...
                        ax(3),'LineStyle','-.');
                    % Create Legend
                    % To create the legend, the first and Second line
will
                    % be plotted on the third axis and then hidden

                    axes(ax(3))

dummy1=line(X_Data,Y_Data1,'LineStyle',L1,'Color',C1);
                    %  Will not show dummy line on plot
                    set(dummy1,'Visible','off');

dummy2=line(X_Data,Y_Data2,'LineStyle',L2,'Color',C2);
                    %  Will not show dummy line on plot
                    set(dummy2,'Visible','off');
                    legend('Efficiency','Rotational Speed',...
                        'Shaft Power','Location','Best');
                    set(legend,'Color',[1 1 1])

                    % Make the x-axis line that lies between the second
                    % and third axes white so it does not appear when
                    % copying figure to past into Word
                    line([xlim1(2) xlim3(2)],[ylim3(1) ylim3(1)],...
                        'Color',[1 1
1],'Parent',ax(3),'Clipping','off');

                    % Create Labels
                    set(get(ax(1),'ylabel'),'string',...
                        'Rotational Speed, \omega [rad/s]')
                    set(get(ax(2),'ylabel'),'string',...
                        'Shaft Power, P_S_h_a_f_t [W]')
                    set(get(ax(3),'ylabel'),'string',...
                        'Efficiency, \eta_m')

                    % Uncomment the following lines to see two voltages
                    % plotted on the same plot.  V2 must be less than
V1
                    % to see the entire plot.
%                     V2=12;
%                     Q_Stall2=(V2-I0*R)/(Kv*R);
%                     I_Stall2=(Q_Stall2*Kv)+I0;
%
%                     if LL==1
%                         fig=12;
%                         EM_I_Range_Large2=I0:.1:I_Stall2;
%                     end
%                     if LL==2
%                         fig=13;
%                         EM_I_Range_Large2=I0:.1:Imax;
```

```
%                       end
%
%                       EM_w_Range_V2=(V2-EM_I_Range_Large2*R)*Kv;
%                       EM_Q_Range_V2=(EM_I_Range_Large2-I0)/Kv;
%                       EM_P_Range_V2=EM_w_Range_V2.*EM_Q_Range_V2;
%
EM_Eff_Range_V2=EM_P_Range_V2./(EM_I_Range_Large2*V2);
%
%                       X_Data_2=EM_Q_Range_V2;
%                       Y_Data1_2=EM_w_Range_V2;
%                       Y_Data2_2=EM_P_Range_V2;
%                       Y_Data3_2=EM_Eff_Range_V2;
%
%                       axes(ax(1))
%                       hold on
%                       H12= plot(X_Data_2,Y_Data1_2,'LineStyle','-',...
%                           'Color','k','LineWidth',1.5);
%                       axes(ax(2))
%                       hold on
%                       H22= plot(X_Data_2,Y_Data2_2,'LineStyle','--',...
%                           'Color','b','LineWidth',1.5);
%                       axes(ax(3))
%                       hold on
%                       H32 = line(X_Data_2,Y_Data3_2,'Color','r',...
%                          'Parent',ax(3),'LineStyle','-
.','LineWidth',1.5);

                  end
                  end
              end
              %%
              if Plot_Switch==1;
                  if Test_Case==1 || Test_Case==2 && ii~=1 && ij~=1
                  %Plot Torque vs Efficiency
                  figure (14)
                  ax1=gca;
                  p1=plot(EM_Current_vs_volt,EM_Eff_vs_volt*100);
                  grid on
                  xlim([0 max(EM_I_Range)])
                  ylim([0 100])
                  xlabel('Motor Current,I [A]')
                  xlim1 = get(ax1,'XLim');

                  xtickax1=get(ax1,'xtick');
                  xticklabelax1=get(ax1,'xticklabel');
                  xlabel('Motor Current, I [A]');
                  ylabel('Efficiency, \eta_m [%]');
                  xlimax1_2=(xlim1-I0)/Kv;
                  xtickax1_2=(xtickax1-I0)/Kv;

                  % create the legned
                  leg=num2str((EM_V_Range).');
                  [rleg cleg]=size(leg);
                  legend1=legend([leg repmat(' Volts',[rleg 1])]);
                  set(legend,'Color',[1 1 1])
```

```
        end
    end
    %%
    if ii==2 && ij==2

        %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %%% Subplot prop coefficients
        if Plot_Switch==1;
            if Test_Case==2
                figure(15)
                subplot(3,1,1)
                plot(Prop_J_Data,Prop_CT_Data,'k-d',...
                    Prop_J_Data,Prop_CP_Data,'k-*');
                xlabel('Advance Ratio, J (1/rev)')
                ylabel('Coefficients');
                title('Propeller Thrust and Power
Coefficients');
                legend('Thrust Coefficient','Power
Coefficient')
                grid on

                subplot(3,1,2)
                plot(Prop_J_Data,Prop_CQ_Data,'k-^');
                xlabel('Advance Ratio, J (1/rev)')
                ylabel('Torque Coefficient');
                title('Propeller Torque Coefficient');
                grid on

                % Plot prop efficiency
                subplot(3,1,3)
                plot(Prop_J_Data,Prop_Eff_Data*100,'k-o');
                xlabel('Advance Ratio, J (1/rev)')
                ylabel('Efficiency, \eta_m [%]');
                title('Propeller Efficiency');
                grid on
            end
        end
    end
    if Test_Case==2
        % Determine polyfits for Prop Coefficient Plots
        npolyfit=5;
        x1=min(Prop_J_Data):.00001:max(Prop_J_Data);
        % Thrust Coefficient Data
        [Prop_CT_Data_Poly,S1] = polyfit(Prop_J_Data,...
            Prop_CT_Data,npolyfit);

        % used to plot polyfit line
        eval_Prop_CT_Data_Poly = polyval(Prop_CT_Data_Poly,x1);

        % Power Coefficient Data
        [Prop_CP_Data_Poly,S2] = polyfit(Prop_J_Data,...
            Prop_CP_Data,npolyfit);
```

```
% used to plot polyfit line
eval_Prop_CP_Data_Poly = polyval(Prop_CP_Data_Poly,x1);

% Torque Coefficient Data
[Prop_CQ_Data_Poly,S3] = polyfit(Prop_J_Data,...
    Prop_CQ_Data,npolyfit);

% used to plot polyfit line
eval_Prop_CQ_Data_Poly = polyval(Prop_CQ_Data_Poly,x1);
% Propeller Efficiency Data
[Prop_Eff_Data_Poly,S4] = polyfit(Prop_J_Data,...
    Prop_Eff_Data,npolyfit);

% used to plot polyfit line
eval_Prop_Eff_Data_Poly =
polyval(Prop_Eff_Data_Poly,x1);

%%%%Endurance Phase
nD_End=(EM_n_Range/GR*Prop_D);

% This will make sure that J is less than 1 for Vend
nD_End=nD_End(find(nD_End>=Vend));

% Determine J range for propeller at Vend
Prop_J_Range_End=Vend./nD_End;

% Prop speed range for prop with J<1
Prop_n_Range_End=nD_End/Prop_D; %          [rps]

% Prop speed range for prop with J<1
Prop_N_Range_End=Prop_n_Range_End*60;%     [rpm]

% Prop speed range for prop with J<1
Prop_w_Range_End=Prop_N_Range_End*2*pi/60;%[rad/s]

% Propeller Efficiency Range
Prop_Eff_Range_End= polyval(Prop_Eff_Data_Poly,...
    Prop_J_Range_End);

% Propeller CP Range
Prop_CP_Range_End = polyval(Prop_CP_Data_Poly,...
    Prop_J_Range_End);

% Propelelr Shaft Power Range
P_Shaft_Range_End = Prop_CP_Range_End.*(rho.*...
    Prop_n_Range_End.^3.*Prop_D.^5);

% Propeller Power Range
Pend_Range_Check=P_Shaft_Range_End.*Prop_Eff_Range_End;

prop_map_End=[Prop_n_Range_End; Pend_Range_Check;
```

```
                    Prop_Eff_Range_End];

                % Find all the locations in the prop_map_End where the
                % propeller power is close to the required propeller
power,
                % Pend.
                Pend_Find_Upper = Pend + .1;
                Pend_Find_Lower = Pend - .1;
                location_n_End=(find(Pend_Range_Check<=Pend_Find_Upper
...
                    & Pend_Range_Check>=Pend_Find_Lower));

                if EM_Case==1
                    if length(location_n_End)==0
                        disp('')
                        disp('ERROR!')
                        disp('The propeller is not producing enough
power')
                        disp(' with the current gear ratio.')
                        disp('Either decrease the gear ratio or
increase')
                        disp(' the voltage.')
                        disp('')
                        disp('Press Ctrl C to Exit')
                        disp('ERROR!')
                        disp('')
                        pause
                    end
                end

                if length(location_n_End)~=0
                    % Find all the possible propeller speeds for Pend
                    possible_n_End=prop_map_End(:,location_n_End);
                    % Since there may be more than one speed for Pend,
                    % start with the one that is most efficient
                    maxEff_End=max(possible_n_End(3,:));

                    % Find the location in all possible with the max
                    % efficiency

location_location_n_End=(find(possible_n_End(3,:)...
                        ==maxEff_End));

                    % Find the propeller rotational speed
                    % for the best solution

                    n_solution_End=possible_n_End(1,...
                        location_location_n_End);

Pend_Check=possible_n_End(2,location_location_n_End);

                    %  Find the difference between the best possible
                    %  propeller power the required propeller power.
Then
```

```
                    %  determine whether the rotational speed needs to
be
                    %  increased or decreased to force the propeller
power
                    %  to match the required power.
                    Diff_Pend=abs(Pend-Pend_Check);
                    n_delta_End=n_solution_End+.00001;

                    Prop_J_End=Vend/(n_delta_End*Prop_D);
                    Prop_Eff_End =
polyval(Prop_Eff_Data_Poly,Prop_J_End);
                    Prop_CP_End =
polyval(Prop_CP_Data_Poly,Prop_J_End);
                    P_Shaft_End =
Prop_CP_End*(rho*n_delta_End^3*Prop_D^5);
                    Pend_Check=P_Shaft_End*Prop_Eff_End;

                    Diff_Pend2=abs(Pend-Pend_Check);

                    if Diff_Pend2<Diff_Pend
                        direction_End=true;
                    elseif Diff_Pend2>Diff_Pend
                        direction_End=false;
                    end

                    n_Prop_End=n_solution_End;
                    while abs(Pend_Check-Pend)>.00001
                        if direction_End==true
                            n_Prop_End=n_Prop_End+.000001;
                        elseif direction_End==false
                            n_Prop_End=n_Prop_End-.000001;
                        end
                        Prop_J_End=Vend/(n_Prop_End*Prop_D);
                        Prop_Eff_End = polyval(Prop_Eff_Data_Poly,...
                            Prop_J_End);
                        Prop_CP_End = polyval(Prop_CP_Data_Poly,...
                            Prop_J_End);
                        P_Shaft_End = Prop_CP_End*(rho*n_Prop_End^3*...
                            Prop_D^5);
                        Pend_Check=P_Shaft_End*Prop_Eff_End;
                    end

                    %Power
                    Prop_P_End=Pout_EM*Prop_Eff_End;
                    Prop_CP_Range_End= polyval(Prop_CP_Data_Poly,...
                        Prop_J_Range_End);
                    Prop_P_Range_End=Prop_CP_Range_End.*(rho*...
                        Prop_n_Range_End.^3*Prop_D^5)*Prop_Eff_End;

                    %Torque
                    Prop_CQ_Range_End= polyval(Prop_CQ_Data_Poly,...
                        Prop_J_Range_End);
                    Prop_Q_Range_End=Prop_CQ_Range_End.*(rho*...
                        Prop_n_Range_End.^2*Prop_D^5);
```

```
                    Prop_CQ_End=Prop_CP_End/(2*pi);
                    Prop_Q_End=Prop_CP_End/(2*pi)*rho*n_Prop_End^2*...
                        Prop_D^5;

                    %Thrust
                    Prop_T_End=Pend/Vend;
                    Prop_CT_Range_End= polyval(Prop_CT_Data_Poly,...
                        Prop_J_Range_End);
                    Prop_T_Range_End=Prop_CT_Range_End.*(rho*...
                        Prop_n_Range_End.^2*Prop_D^4);
                    Prop_CT_End=Prop_T_End/(rho*n_Prop_End^2*Prop_D^4);

                    %Speed
                    N_Prop_End=n_Prop_End*60;
                    w_Prop_End=n_Prop_End*2*pi;

                    % Set the torque and speed of the EM to the torque
                    % and speed of the prop then run the optimization
                    % routine again

                    if GR_Case==1;
                        Qmax=Prop_Q_End;
                        Qmin=Prop_Q_End;
                        wmin=w_Prop_End;
                        wmax=w_Prop_End;

                    elseif GR_Case==2
                        Qmax=Prop_Q_End/GR;
                        Qmin=Qmax;
                        wmin=w_Prop_End*GR;
                        wmax=w_Prop_End*GR;
                    end
                end
            end
        end
    end

    if Test_Case==2
        %%
        if Plot_Switch==1;
            figure(16)
            plot(Prop_Q_Range_End,Prop_Eff_Range_End*100,'r')
            grid on
            hold on
            plot(Prop_Q_End,Prop_Eff_End*100,'ro','MarkerSize',14,...
                'LineWidth',2.5)
            legend(['V_E_n_d_u_r_a_n_c_e = ',num2str(Vend),'
[m/s]'],...
                ['Q_p = ',num2str(Prop_Q_End),' [N-m]',...
                ', \eta_p = ' num2str(Prop_Eff_End*100),' [%]'])
            ylim([0 100])
            title('Propeller Efficiency vs Torque')
            xlabel('Propeller Torque, Q_p [N-m]')
            ylabel('Efficiency, \eta_p [%]');
```

```
        end
        %%
        if Plot_Switch==1;
            % Plot Propeller Efficiency vs propeller speed range at
Vend
            % and J<1
            figure (17)
            plot(Prop_w_Range_End,Prop_Eff_Range_End*100,'r');
            xlabel('Rotational Speed, \omega [rad/s]')
            ylabel('Efficiency, \eta_p [%]');
            grid on
            hold on
            plot(w_Prop_End,Prop_Eff_End*100,'ro','MarkerSize',14,...
                'LineWidth',2.5)
            legend(['V_E_n_d_u_r_a_n_c_e  = ',num2str(Vend),'
[m/s]'],...
                ['\omega_E_n_d_u_r_a_n_c_e = ',num2str(w_Prop_End),...
                '  [rad/s]',', \eta_p = ' num2str(Prop_Eff_End*100),...
                ' [%]'])
            ylim([0 100])
            title('Efficiency vs Propeller Speed Range at V_E_n_d')
        end
        %%
        if Plot_Switch==1;
            % Plot Motor Torque and Propeller Torque vs speed
            figure (18)
            [III]=find(abs(EM_V_Range-V)<.0001);
            plot(Prop_w_Range_End,Prop_Q_Range_End,'r-','LineWidth',2)
            hold on
            plot(EM_w_Range,EM_Torque_vs_volt(:,III),'k--
','LineWidth',2)
            xlabel('Rotational Speed, \omega [rad/s]')
            ylabel('Torque,Q [N-m]');
            grid on
            if GR~=1
                plot(EM_w_Range/GR,EM_Torque_vs_volt(:,III)*GR,'b:',...
                    'LineWidth',2);
            end
            plot(w_Prop_End,Prop_Q_End,'r^','MarkerSize',14,...
                'LineWidth',2.5)
            ylim([0 max(EM_Q_Range)*GR])
            title(['Torque vs Rotational Speed at V_E_n_d =',...
                num2str(Vend),' [m/s]'])
            if GR~=1
                legend('Propeller','Electric Motor','Shaft',...
                    ['\omega = ',num2str(w_Prop_End),...
                    ' [rad/s]',', Q = ',num2str(Prop_Q_End),' [N-m]'])
            else
                legend('Propeller','Electric Motor',['\omega = ',...
                    num2str(w_Prop_End),' [rad/s]',', Q = ',...
                    num2str(Prop_Q_End),' [N-m]'])
            end
        end
        %%
```

```
        %%%%%%%
        %%%%Cruise Phase
        nD_Cruise=(ICE_n_Range)*Prop_D;

        % This will make sure that J is less than 1 for Vcruise
        nD_Cruise=nD_Cruise(find(nD_Cruise>=Vcruise));

        % Determine J range for propeller at Vcruise
        Prop_J_Range_Cruise=Vcruise./nD_Cruise;

        % Prop speed range for prop with J<1
        Prop_n_Range_Cruise=nD_Cruise/Prop_D; % [rps]

        % Prop speed range for prop with J<1
        Prop_N_Range_Cruise=Prop_n_Range_Cruise*60;% [rpm]

        % Prop speed range for prop with J<1
        Prop_w_Range_Cruise=Prop_N_Range_Cruise*2*pi/60;%[rad/s]

        %%%%%%
        % Propelelr Efficiency Range
        Prop_Eff_Range_Cruise= polyval(Prop_Eff_Data_Poly,...
            Prop_J_Range_Cruise);

        % Propeller CP Range
        Prop_CP_Range_Cruise = polyval(Prop_CP_Data_Poly,...
            Prop_J_Range_Cruise);

        % Propelelr Shaft Power Range
        P_Shaft_Range_Cruise = Prop_CP_Range_Cruise.*...
            (rho.*Prop_n_Range_Cruise.^3.*Prop_D.^5);

        % Propeller Power Range

Pcruise_Range_Check=P_Shaft_Range_Cruise.*Prop_Eff_Range_Cruise;

        prop_map_Cruise=[Prop_n_Range_Cruise; Pcruise_Range_Check;
            Prop_Eff_Range_Cruise];

        % Find all the locations in the prop_map_Cruise where the
propeler
        % power is close to the required propeller power, Pcruise.
        Pcruise_Find_Upper = Pcruise + .1;
        Pcruise_Find_Lower = Pcruise - .1;
        location_n_Cruise=(find(Pcruise_Range_Check<=Pcruise_Find_Upper
...
            & Pcruise_Range_Check>=Pcruise_Find_Lower));

        % Find all the possible propeller speeds for Pcruise
        possible_n_Cruise=prop_map_Cruise(:,location_n_Cruise);
        % Since there may be more than one speed for Pcruise, start
with
        % the one that is most efficient
```

```
        maxEff_Cruise=max(possible_n_Cruise(3,:));

        % Find the location in all possible with the max efficiency
        location_location_n_Cruise=(find(possible_n_Cruise(3,:)...
            ==maxEff_Cruise));

        % Find the propeller rotational speed for the
        % best solution


n_solution_Cruise=possible_n_Cruise(1,location_location_n_Cruise);
        Pcruise_Check=possible_n_Cruise(2,location_location_n_Cruise);

        %  Find the difference between the best possible propeller
        %  power the required propeller power.  Then determine whether
        %  the rotational speed needs to be increased or decreased to
        %  force the propeller power to match the required power.
        Diff_Pcruise=abs(Pcruise-Pcruise_Check);
        n_delta_Cruise=n_solution_Cruise+.00001;

        Prop_J_Cruise=Vcruise/(n_delta_Cruise*Prop_D);
        Prop_Eff_Cruise = polyval(Prop_Eff_Data_Poly,Prop_J_Cruise);
        Prop_CP_Cruise = polyval(Prop_CP_Data_Poly,Prop_J_Cruise);
        P_Shaft_Cruise =
Prop_CP_Cruise*(rho*n_delta_Cruise^3*Prop_D^5);
        Pcruise_Check=P_Shaft_Cruise*Prop_Eff_Cruise;

        Diff_Pcruise2=abs(Pcruise-Pcruise_Check);

        if Diff_Pcruise2<Diff_Pcruise
            direction_Cruise=true;
        elseif Diff_Pcruise2>Diff_Pcruise
            direction_Cruise=false;
        end

        n_Prop_Cruise=n_solution_Cruise;
        while abs(Pcruise_Check-Pcruise)>.00001
            if direction_Cruise==true
                n_Prop_Cruise=n_Prop_Cruise+.000001;
            elseif direction_Cruise==false
                n_Prop_Cruise=n_Prop_Cruise-.000001;
            end
            Prop_J_Cruise=Vcruise/(n_Prop_Cruise*Prop_D);
            Prop_Eff_Cruise =
polyval(Prop_Eff_Data_Poly,Prop_J_Cruise);
            Prop_CP_Cruise = polyval(Prop_CP_Data_Poly,Prop_J_Cruise);
            P_Shaft_Cruise =
Prop_CP_Cruise*(rho*n_Prop_Cruise^3*Prop_D^5);
            Pcruise_Check=P_Shaft_Cruise*Prop_Eff_Cruise;
        end

        %Power
        Prop_P_Cruise=Prop_Eff_Cruise*P_Shaft_Cruise;
```

```
        Prop_CP_Range_Cruise= polyval(Prop_CP_Data_Poly,...
            Prop_J_Range_Cruise);
        Prop_P_Range_Cruise=Prop_CP_Range_Cruise.*...
            (rho*Prop_n_Range_Cruise.^3*Prop_D^5)*Prop_Eff_Cruise;

        %Torque
        Prop_CQ_Range_Cruise= polyval(Prop_CQ_Data_Poly,...
            Prop_J_Range_Cruise);
        Prop_Q_Range_Cruise=Prop_CQ_Range_Cruise.*...
            (rho*Prop_n_Range_Cruise.^2*Prop_D^5);
        Prop_CQ_Cruise=Prop_CP_Cruise/(2*pi);

Prop_Q_Cruise=Prop_CP_Cruise/(2*pi)*rho*n_Prop_Cruise^2*Prop_D^5;

        %Thrust
        Prop_T_Cruise=Pcruise/Vcruise;
        Prop_CT_Range_Cruise= polyval(Prop_CT_Data_Poly,...
            Prop_J_Range_Cruise);
        Prop_T_Range_Cruise=Prop_CT_Range_Cruise.*...
            (rho*Prop_n_Range_Cruise.^2*Prop_D^4);
        Prop_CT_Cruise=Prop_T_Cruise/(rho*n_Prop_Cruise^2*Prop_D^4);

        %Speed
        N_Prop_Cruise=n_Prop_Cruise*60;
        w_Prop_Cruise=n_Prop_Cruise*2*pi;

        if w_Prop_Cruise>Prop_w_Range_Cruise
            disp('')
            disp('ERROR!')
            disp('The required propeller rotation rate for cruise
exceeds')
            disp(' the maximum propeller rotation speed.  Choose a
larger')
            disp(' propeller.')
            disp('')
            disp('Press Ctrl C to Exit')
            disp('ERROR!')
            disp('')
            disp('')
            pause
        end



        %%
        % Set the voltage range greater than normal voltage range to
        % produce an increased speed and efficiency range
        VVV=0:max(Vmax,60);%                         [V]     Voltage
Range
        EMRNG=Kv*((R*I0:.01:max(Vmax,60))-R*I0);%    [rad/s] Speed
Range

        % Efficiency as a function of voltage (columns) and speed(rows)
        for ctr1=1:length(VVV)
            EEE(:,ctr1)=(1-(I0*R./(VVV(ctr1)-(EMRNG./Kv)))).*...
```

```
        (EMRNG/(VVV(ctr1)*Kv));
end
% make sure efficiency is greater than 0 and less than 1
EEE(EEE<0)=0;
EEE(EEE>1)=0;
% Max_Eff_Volt reterns the max efficiency of each voltage
% Row returnes the the row location of each max efficiency
[Max_Eff_Volt Row]=max(EEE);
% Sp returns the speed associated with each max efficiency
sp=EMRNG(Row);
%%
if Plot_Switch==1;
    figure(19)
    plot(EM_w_Range,EM_Eff_vs_volt*100)
    hold on
    plot(sp,Max_Eff_Volt*100)
    grid on
    xlabel('Rotational Speed, \omega [rad/s]');
    ylabel('Max Efficiency, \eta_m [%]');
    clear VVV
end



%%
% Generator Efficiency at Cruise Speed

w_Gen=w_Prop_Cruise*GR;
[EFF_vs_Speed_Poly, S5] = polyfit(sp,Max_Eff_Volt,npolyfit);
Gen_Eff=polyval(EFF_vs_Speed_Poly,w_Gen);
V_Gen=((w_Gen^2-4*PGen*Kv^2*R)^.5+w_Gen)/(2*Kv);
I_Gen=PGen/V_Gen;
Pout_Gen=I_Gen*V_Gen;
Q_Gen=(I_Gen+I0)/Kv;
Pin_Gen=Q_Gen*w_Gen;
Eff_Gen=I_Gen*V_Gen/(Pin_Gen);

% Power generation during cruise
% ICE Power for Cruise
Q_ICE_Cruise=Prop_Q_Cruise/Eff_Clutch;
P_ICE_Cruise=Q_ICE_Cruise*w_Prop_Cruise;

% ICE Power for Regeneration
Q_ICE_Gen=Q_Gen/Eff_Clutch;
P_ICE_Gen=Q_ICE_Gen*w_Gen;

% ICE Power Combined cruise and regeneration
Q_ICE_Cruise_Gen=Q_ICE_Cruise+Q_ICE_Gen;
P_ICE_Cruise_Gen=P_ICE_Cruise+P_ICE_Gen;

%%
if Plot_Switch==1;
    % Plot Thrust Coefficient and poly fit
    figure (20)
    plot(Prop_J_Data,Prop_CT_Data,'k-d')
```

```
        grid on
        hold on
        plot(x1,eval_Prop_CT_Data_Poly)
        xlabel('Advance Ratio, J (1/rev)')
        ylabel('Thrust Coefficient, C _T');
        legend('Data','Polyfit')
        plot(Prop_J_End,Prop_CT_End,'rd','MarkerSize',14,...
            'LineWidth',2.5)
        plot(Prop_J_Cruise,Prop_CT_Cruise,'bd','MarkerSize',14,...
            'LineWidth',2.5)
        legend('Data','Curve Fit', 'Endurance','Cruise')
    end

    %%
    if Plot_Switch==1;
        % Plot Power Coefficient and poly fit
        figure(21)
        plot(Prop_J_Data,Prop_CP_Data,'k-*');
        grid on
        hold on
        plot(x1,eval_Prop_CP_Data_Poly)
        xlabel('Advance Ratio, J (1/rev)')
        ylabel('Power Coefficient, C _P');
        plot(Prop_J_End,Prop_CP_End,'r*','MarkerSize',14,...
            'LineWidth',2.5)
        plot(Prop_J_Cruise,Prop_CP_Cruise,'b*','MarkerSize',14,...
            'LineWidth',2.5)
        legend('Data','Curve Fit', 'Endurance','Cruise')
    end

    %%
    if Plot_Switch==1;
        % Plot Torque Coefficient and poly fit
        figure (22)
        plot(Prop_J_Data,Prop_CQ_Data,'k-
^',x1,eval_Prop_CQ_Data_Poly);
        xlabel('Advance Ratio, J (1/rev)')
        ylabel('Torque Coefficient, C _Q');
        grid on
        hold on
        plot(Prop_J_End,Prop_CQ_End,'r^','MarkerSize',14,...
            'LineWidth',2.5)
        plot(Prop_J_Cruise,Prop_CQ_Cruise,'b^','MarkerSize',14,...
            'LineWidth',2.5)
        legend('Data','Curve Fit', 'Endurance','Cruise')
    end

    %%
    if Plot_Switch==1;
        % Plot Efficiency and poly fit
        figure (23)
        plot(Prop_J_Data,Prop_Eff_Data*100,'k-o',x1,...
            eval_Prop_Eff_Data_Poly*100);
        xlabel('Advance Ratio, J (1/rev)')
        ylabel('Efficiency, \eta_p');
```

```
            grid on
            hold on
            plot(Prop_J_End,Prop_Eff_End*100,'ro','MarkerSize',14,...
                'LineWidth',2.5)

plot(Prop_J_Cruise,Prop_Eff_Cruise*100,'bo','MarkerSize',...
                14,'LineWidth',2.5)
            legend('Data','Curve Fit', 'Endurance','Cruise')
            axis([0,1,0,100])
        end
        %%
        if Plot_Switch==1;
            figure(24)
            plot(Prop_Q_Range_Cruise,Prop_Eff_Range_Cruise*100,'b')
            grid on
            hold on

plot(Prop_Q_Cruise,Prop_Eff_Cruise*100,'bo','MarkerSize',14,...
                'LineWidth',2.5)
            legend(['V_C_r_u_i_s_e [m/s] = ',num2str(Vcruise)],...
                ['Q_p [N-m] = ',num2str(Prop_Q_Cruise),...
                ', \eta_p [%] = ' num2str(Prop_Eff_Cruise*100)])
            ylim([0 100])
            title('Propeller Efficiency vs Torque')
            xlabel('Propeller Torque, Q_p [N-m]')
            ylabel('Efficiency, \eta_p [%]');
        end
        %%
        if Plot_Switch==1;
            figure(25)
            title('Propeller Efficiency vs Torque')
            plot(Prop_Q_Range_End,Prop_Eff_Range_End*100,'r')
            grid on
            hold on
            plot(Prop_Q_End,Prop_Eff_End*100,'ro','MarkerSize',14,...
                'LineWidth',2.5)
            plot(Prop_Q_Range_Cruise,Prop_Eff_Range_Cruise*100,'b')
            grid on
            hold on

plot(Prop_Q_Cruise,Prop_Eff_Cruise*100,'bo','MarkerSize',14,...
                'LineWidth',2.5)
            legend(['V_E_n_d_u_r_a_n_c_e [m/s] = ',num2str(Vend)],...
                ['Q_p [N-m] = ',num2str(Prop_Q_End),...
                ', \eta_p = ' num2str(Prop_Eff_End*100),' [%]'],...
                ['V_C_r_u_i_s_e = ',num2str(Vcruise),' [m/s]'],...
                ['Q_p = ',num2str(Prop_Q_Cruise),...
                ' [N-m]',', \eta_p = '
num2str(Prop_Eff_Cruise*100),'[%]'])
            ylim([0 100])
            title('Propeller Efficiency vs Torque')
            xlabel('Propeller Torque, Q_p [N-m]')
            ylabel('Efficiency, \eta_p [%]');
        end
        %%
```

```
        if Plot_Switch==1;
            % Plot Propeller Efficiency vs propeller speed range at
            % Vcruise and J<1
            figure (26)
            plot(Prop_w_Range_Cruise,Prop_Eff_Range_Cruise*100,'b');
            xlabel('Rotational Speed, \omega [rad/s]')
            ylabel('Efficiency, \eta_p [%]');
            grid on
            hold on

plot(w_Prop_Cruise,Prop_Eff_Cruise*100,'bo','MarkerSize',...
                14,'LineWidth',2.5)
            legend(['V_C_r_u_i_s_e = ',num2str(Vcruise),' [m/s] '],...
                ['\omega_C_r_u_i_s_e = ',num2str(w_ICE),...
                ' [rad/s]',', \eta_p = '
num2str(Prop_Eff_Cruise*100),...
                ' [%] '])
            ylim([0 100])
            title('Propeller Efficiency vs Speed Range at Vcruise')
        end
        %%
        if Plot_Switch==1;
            figure (27)
            title('Propeller Efficiency vs Propeller Speed')
            plot(Prop_w_Range_End,Prop_Eff_Range_End*100,'r');
            grid on
            hold on
            plot(w_Prop_End,Prop_Eff_End*100,'ro','MarkerSize',14,...
                'LineWidth',2.5)
            plot(Prop_w_Range_Cruise,Prop_Eff_Range_Cruise*100,'b');

plot(w_Prop_Cruise,Prop_Eff_Cruise*100,'bo','MarkerSize',14,...
                'LineWidth',2.5)
            ylim([0 100])
            xlabel('Rotational Speed, \omega [rad/s]')
            ylabel('Efficiency, \eta_p [%]');
            legend(['V_E_n_d_u_r_a_n_c_e  = ',num2str(Vend),'
[m/s]'],...
                ['\omega_E_n_d_u_r_a_n_c_e = ',num2str(w_Prop_End),...
                '  [rad/s]',', \eta_p = ' num2str(Prop_Eff_End*100),...
                ' [%] '],['V_C_r_u_i_s_e = ',num2str(Vcruise),...
                ' [m/s] '],['\omega_C_r_u_i_s_e = ',num2str(w_ICE),...
                ' [rad/s]',', \eta_p = '
num2str(Prop_Eff_Cruise*100),...
                ' [%] '])
        end
        %%
        if Plot_Switch==1;
            % Big Picture plot
            figure(28)
            subplot(4,1,1)
            plot(Prop_w_Range_End,Prop_T_Range_End);
            hold on
            plot(w_Prop_End,Prop_T_End,'ro','MarkerSize',14,...
                'LineWidth',2.5)
```

```
            plot(0:10:w_Shaft_End,Prop_T_End,'r>','MarkerSize',3);
            plot(w_Shaft_End,Prop_T_End:-5:0,'rv','MarkerSize',3)
            grid on
            xlim([min(EM_w_Range) max(EM_w_Range)])

            ylim([0 Prop_T_End*2])
            %     xlabel('Rotational Speed, \omega [rad/s]')
            ylabel({'Propeller Thrust,','T [N]'})
            legend(['V_E_n_d_u_r_a_n_c_e = ',num2str(Vend),'
[m/s]'],...
                ['\omega = ',num2str(w_Prop_End),...
                ' [rad/s]',', T = ' num2str(Prop_T_End),' [N]'])

            subplot(4,1,2)
            if abs(EM_V_Range(III)-Vmax)>5
                [JJJ]=[EM_V_Range(III-1),EM_V_Range(III),...
                    EM_V_Range(III+1)];
            else
                [JJJ]=[EM_V_Range(III-1),EM_V_Range(III)];
            end

            for ctr=1:length(JJJ)
                EM_T_V(:,ctr)=((JJJ(ctr)-(EM_w_Range./Kv))/R-I0)/Kv;
                EM_E_V(:,ctr)=(1-(I0*R./(JJJ(ctr)-
(EM_w_Range./Kv)))).*...
                    (EM_w_Range/(JJJ(ctr)*Kv));
            end
            EM_E_V(EM_E_V<0)=0;
            EM_E_V(EM_E_V>1)=0;

            plot(EM_w_Range/GR,EM_T_V*GR)
            hold on
            plot(Prop_w_Range_End,Prop_Q_Range_End,'k')
            plot(w_Prop_End,Q_Shaft_End,'ro','MarkerSize',14,...
                'LineWidth',2.5)
            plot(w_Shaft_End,0:max(EM_Q_Range)/10:max(EM_Q_Range),...
                'rv','MarkerSize',3)
            ylim([0 max(EM_Q_Range)*GR])
            xlim([min(EM_w_Range) max(EM_w_Range)])
            grid on
            %     xlabel('Rotational Speed, \omega [rad/s]')
            ylabel({'Torque,','Q [N-m]'})
            if abs(EM_V_Range(III)-Vmax)>5
                legend(['Shaft Torque at ',num2str(JJJ(1)),'
Volts'],...
                    ['Shaft Torque at ',num2str(JJJ(2)),' Volts'],...
                    ['Shaft Torque at ',num2str(JJJ(3)),' Volts'],...
                    'Propeller Torque',['\omega [rad/s] = ',...
                    num2str(w_Prop_End),...
                    ', Q [N-m] = ' num2str(Q_Shaft_End)])
            else
                legend(['Shaft Torque at ',num2str(JJJ(1)),'
Volts'],...
                    ['Shaft Torque at ',num2str(JJJ(2)),' Volts'],...
```

```
                    'Propeller Torque',['\omega =
',num2str(w_Prop_End),...
                    ' [rad/s]',', Q = ' num2str(Q_Shaft_End),' [N-m]'])
            end
            subplot(4,1,3)
            plot(Prop_w_Range_End,Prop_Eff_Range_End*100);
            %     xlabel('Rotational Speed, \omega [rad/s]')
            ylabel({'Propeller Efficiency,',' \eta_p [%]'});
            grid on
            hold on
            plot(w_Prop_End,Prop_Eff_End*100,'ro','MarkerSize',14,...
                'LineWidth',2.5)
            plot(w_Shaft_End,0:10:100,'rv','MarkerSize',3)
            legend(['V_E_n_d_u_r_a_n_c_e = ',num2str(Vend),...
                ' [m/s]', ', P_E_n_d_u_r_a_n_c_e = ',num2str(Pend),...
                ' [W]'],['\omega = ',num2str(w_Prop_End),...
                ' [rad/s]',', \eta_p = ' num2str(Prop_Eff_End*100),'
[%]'])
            ylim([0 100])
            xlim([min(EM_w_Range) max(EM_w_Range)])

            subplot(4,1,4)
            plot(EM_w_Range,EM_E_V*100)
            hold on
            plot(w_EM,EM_Eff*100,'ro','MarkerSize',14,'LineWidth',2.5)
            plot(w_EM,100:-5:EM_Eff*100,'rv','MarkerSize',3)
            ylim([0 100])
            xlim([min(EM_w_Range)*GR max(EM_w_Range)*GR])
            grid on
            xlabel('Rotational Speed, \omega [rad/s]')
            ylabel({'Electric Motor Efficiency,',' \eta_m [%]'})
            if abs(EM_V_Range(III)-Vmax)>5
                legend([num2str(JJJ(1)),' Volts'],[num2str(JJJ(2)),...
                    ' Volts'],[num2str(JJJ(3)),' Volts'],...
                    ['\omega = ',num2str(w_EM),...
                    ' [rad/s]',', \eta_m = ' num2str(EM_Eff*100),'
[%]'])
            else
                legend([num2str(JJJ(1)),' Volts'],[num2str(JJJ(2)),...
                    ' Volts'],['\omega = ',num2str(w_EM),...
                    '[rad/s]',', \eta_m = ' num2str(EM_Eff*100),
'[%]'])
            end
        end
        clear EM_Current_vs_volt EM_Torque_vs_volt EM_Power_vs_volt
        clear EM_Eff_vs_volt EEE EM_T_V EM_E_V


    end
    if ctrR<=length(R_Range)
        Output_R_Range(ctrR,:)=[R I0 V I Kv EM_Eff Qmin Qmax wmin
wmax];
        if EM_Case==2
            waitbar(ctrR/length(R_Range),h_wait)
        end
```

```
        if ctrR==length(R_Range)
            Output_R_Range2=Output_R_Range((abs(Output_R_Range(:,7)-...
                Qmin)<=0.0001),:);
            Output_R_Range3=Output_R_Range2((abs(Output_R_Range2(:,9)-
...
                wmin)<=0.0001),:);
            Output_R_Range4=Output_R_Range3((Output_R_Range3(:,3)...
                <=Vmax+.00001),:);
            Output_R_Range5=Output_R_Range4((Output_R_Range4(:,4)...
                <=Imax),:);
            Output_R_Range6=Output_R_Range5((Output_R_Range5(:,6)...
                ==max(Output_R_Range5(:,6))),:);
            R=Output_R_Range6(1);
            I0=Output_R_Range6(2);
            % V=Output_R_Range6(3);
            % I=Output_R_Range6(4);
            Kvmin=Output_R_Range6(5);
            % EM_Eff=Output_R_Range6(6);
            Kvmax=Kvmin;
        end
    end
end
%%
clc
% Display to screen
% Title and Date-Time Stamp
disp('Capt Todd Rotramel (USAF)')
disp('Air Force Institute Technology')
disp('Masters Thesis: Optimization of Propeller-Based Hybrid-Electric')
disp('              Propulsion System for Small Remotely-Piloted
Aircraft')

disp(' ');
disp(' ');
timestamp = clock;
disp('Hybrid-Electric RPA Component Sizing Program');
disp(['Date: ',date,'      Time: ',num2str(timestamp(4)),...
    ':', num2str(timestamp(5))]);

if Test_Case==2
    disp(' ');
    disp('Altitudes ');
    disp(['Take-off Altitude  MSL      [m]       = ', num2str(h_TO)]);
    disp(['Mission  Altitude  AGL      [m]       = ', num2str(h_AGL)]);
    disp(['Mission Altitude Density    [kg/m^3]  = ', num2str(rho)]);

    %Prop
    disp(' ');
    disp('Propeller');
    disp(Prop_Brand)
    disp(['Diameter                    [in]      = ',
num2str(Prop_Dia)]);
    disp(['Diameter                    [m]       = ',
num2str(Prop_D)]);
    disp(['Pitch                       [in]      = ',...
```

```
        num2str(Prop_Pitch)]);
    disp(['Pitch                       [m]       = ',...
        num2str(Prop_Pitch*0.0254)])

    %ICE
    disp(' ');
    disp('Internal Combustion Engine');
    disp(['Max ICE Speed               [rad/s]   = ',
num2str(wmax_ICE)]);
    disp(['Max ICE Speed               [rpm]     = ',
num2str(Nmax_ICE)]);
    disp(['Min ICE Speed               [rad/s]   = ',
num2str(wmin_ICE)]);
    disp(['Min ICE Speed               [rpm]     = ',
num2str(Nmin_ICE)]);

    %Generator
    disp(' ');
    disp('Generator');
    disp(['Min Generator Speed         [rad/s]   = ',
num2str(wmin_GEN)]);
    disp(['Min Generator Speed         [rpm]     = ',
num2str(Nmin_GEN)]);

end
%Battery
disp(' ');
disp('Battery');
disp(['Single Battery Voltage      [Volts]   = ', num2str(V_bat)]);
disp(['Single Battery Capacity     [A-h]     = ', num2str(C_bat)]);
disp(['Number of Batteries         [ea]      = ', num2str(Num_bat)]);
disp(['Total Battery Capacity      [A-h]     = ', num2str(C_Total)]);

%DC/DC convertor
disp(' ');
disp('DC/DC Convertor');
disp(['Max Current                 [Amps ]   = ', num2str(Imax)]);
disp(['Max Battery Voltage         [Volts]   = ', num2str(Vmax)]);

%Gear Ratio
disp(' ');
disp(['Gear Ratio                  [EM/ICE]  =',num2str(rats(GR,5))]);
disp(['Gear Ratio                  [EM/ICE]  = ', num2str(GR)])

%Electric Motor
disp(' ');
disp('Electric Motor');
disp(['EM Internal Resistance      [Ohms]    = ', num2str(R)]);
disp(['No-load Current             [Amps]    = ', num2str(I0)]);
disp(['Motor Speed Constant        [rad/s/V] = ', num2str(Kv)]);
disp(['Motor Speed Constant        [rpm/V]   = ', num2str(Kv_N)]);
disp(['Max Continuous Torque       [N-m]     = ', num2str(Qmax_EM)]);

%Endurance
```

```
disp(' ');
disp('Endurance');
disp(['Endurance Speed              [m/s]     = ', num2str(Vend)]);
disp(['Endurance Time               [sec]     = ', num2str(tend)]);
disp(['Endurance Time               [hr]      = ', num2str(tend/3600)]);
disp(['EM Speed                     [rad/s]   = ', num2str(w_EM)]);
disp(['EM Speed                     [rpm]     = ', num2str(N_EM)]);
disp(['Required Voltage             [Volts]   = ', num2str(V)]);
disp(['Required Current             [Amps]    = ', num2str(I)]);
disp(['Starting Current             [Amps]    = ', num2str(I_Stall)]);
disp(['Most Eff Current @ V Req     [Amps]    = ', num2str(I_maxEff)]);
disp(['Most Eff EM Speed @ V Req    [rad/s]   = ', num2str(w_maxEff)]);
disp(['Most Eff EM Speed @ V Req    [rpm]     = ', num2str(N_maxEff)]);
disp(['Stall Torque                 [N-m]     = ', num2str(Q_Stall)]);
disp(['EM Torque                    [N-m]     = ', num2str(Q_EM)]);
disp(['No-Load Speed                [rad/s]   = ', num2str(w0)]);
disp(['No-Load Speed                [rpm]     = ', num2str(N0)]);
disp(['EM Input Power               [W]       = ', num2str(Pin_EM)]);
disp(['EM Output Power              [W]       = ', num2str(Pout_EM)]);
disp(['EM Efficiency                [%]       = ',
num2str(EM_Eff*100)]);
disp(['Max EM Eff @ V Req           [%]       = ',
num2str(Effmax_EM*100)]);
if Test_Case==2
    disp(['Max Shaft Torque             [N-m]     = ',...
        num2str(Qmax_Shaft)]);
    disp(['Shaft Torque                 [N-m]     = ',...
        num2str(Q_Shaft_End)]);
    disp(['Shaft Speed                  [rad/s]   = ',...
        num2str(w_Shaft_End)]);
    disp(['Shaft Speed                  [rpm]     = ',...
        num2str(w_Shaft_End)]);
    disp(['Shaft Power                  [W]       = ',...
        num2str(P_Shaft_End)]);
    disp(['Endurance Prop Speed         [rad/s]   = ',...
        num2str(w_Prop_End)]);
    disp(['Endurance Prop Speed         [rpm]     = ',...
        num2str(N_Prop_End)]);
    disp(['Endurance Advance Ratio      []        = ',...
        num2str(Prop_J_End)]);
    disp(['Endurance Prop Thrust        [N]       = ',...
        num2str(Prop_T_End)]);
    disp(['Endurance Prop Power         [W]       = ',...
        num2str(Prop_P_End)]);
    disp(['Endurance Prop Torque        [N-m]     = ',...
        num2str(Prop_Q_End)]);
    disp(['Endurance Prop Efficiency    [%]       = ',...
        num2str(Prop_Eff_End*100)]);
    disp(['Endurance Efficiency         [%]       = ',...
        num2str(Prop_Eff_End*EM_Eff*100)]);

    %Cruise
    disp(' ');
    disp('Cruise');
```

```
    disp(['Cruise Speed                [m/s]     = ',
num2str(Vcruise)]);
    disp(['ICE Speed                   [rad/s]   = ',...
        num2str(w_Prop_Cruise)]);
    disp(['ICE Speed                   [rpm]     = ',...
        num2str(N_Prop_Cruise)]);
    disp(['ICE Output Power            [W]       = ',...
        num2str(P_ICE_Cruise)]);
    disp(['ICE Output Torque           [N-m]     = ',...
        num2str(Q_ICE_Cruise)]);
    disp(['Shaft Torque                [N-m]     = ',...
        num2str(Prop_Q_Cruise)]);
    disp(['Shaft Speed                 [rad/s]   = ',...
        num2str(w_Prop_Cruise)]);
    disp(['Shaft Speed                 [rpm]     = ',...
        num2str(N_Prop_Cruise)]);
    disp(['Shaft Power                 [W]       = ',...
        num2str(P_Shaft_Cruise)]);
    disp(['Cruise Prop Speed           [rad/s]   = ',...
        num2str(w_Prop_Cruise)]);
    disp(['Cruise Prop Speed           [rpm]     = ',...
        num2str(N_Prop_Cruise)]);
    disp(['Cruise Advance Ratio        []        = ',...
        num2str(Prop_J_Cruise)]);
    disp(['Cruise Prop Thrust          [N]       = ',...
        num2str(Prop_T_Cruise)]);
    disp(['Cruise Prop Power           [W]       = ',...
        num2str(Prop_P_Cruise)]);
    disp(['Cruise Prop Torque          [N-m]     = ',...
        num2str(Prop_Q_Cruise)]);
    disp(['Cruise Prop Efficiency      [%]       = ',...
        num2str(Prop_Eff_Cruise*100)]);

    %Cruise Plus Regeneration
    disp(' ');
    disp('Cruise plus Regeneration');
    disp(['Generator Output Power      [W]       = ',...
        num2str(Pout_Gen)]);
    disp(['Generator Speed             [rad/s]   = ',...
        num2str(w_Gen)]);
    disp(['Generator Speed             [rpm]     = ',...
        num2str(N_Prop_Cruise*GR)]);
    disp(['Generator Required Torque   [N-m]     = ',...
        num2str(Q_Gen)]);
    disp(['Generator Input Power       [W]       = ',...
        num2str(Pin_Gen)]);
    disp(['Generator Output Voltage    [V]       = ',...
        num2str(V_Gen)]);
    disp(['Generator Output Current    [A]       = ',...
        num2str(I_Gen)]);
    disp(['Generator Efficiency        [%]       = ',...
        num2str(Eff_Gen*100)])
    disp(['ICE Output Power            [W]       = ',...
        num2str(P_ICE_Cruise_Gen)]);
    disp(['ICE Output Torque           [N-m]     = ',...
```

```
        num2str(Q_ICE_Cruise_Gen)]);
    if EM_Case==2; close(h_wait); end
end
disp(' ');
toc
end

function [f]=Design_obj(x)
global R I0

f=-(((x(3)-I0)*(x(1)/x(2)))/(((x(1)/x(2))+(x(3)*R))*x(3)));

end

function [C,Ceq]=Design_const1(x)
global R I0 wmax wmin Kvmax Kvmin Imax Imin Vmax Vmin Qmax Qmin

C=[x(1)-wmax
    -x(1)+wmin
    x(2)-Kvmax
    -x(2)+Kvmin
    x(3)-Imax;
    -x(3)+Imin
    (x(1)/x(2)+x(3)*R)-Vmax
    -(x(1)/x(2)+x(3)*R)+Vmin
    (x(3)-I0)/x(2)-Qmax
    -((x(3)-I0)/x(2))+Qmin];
Ceq=[];
end

function [C,Ceq]=Design_const2(x)
global R I0 wmax wmin Kvmax Kvmin Imax Imin Vmax Vmin Qmax Qmin GRmax
GRmin

C=[x(1)-wmax*x(4)
    -x(1)+wmin*x(4)
    x(2)-Kvmax
    -x(2)+Kvmin
    x(3)-Imax
    -x(3)+Imin
    (x(1)/x(2)+x(3)*R)-Vmax
    -(x(1)/x(2)+x(3)*R)+Vmin
    (x(3)-I0)/x(2)-Qmax/x(4)
    -((x(3)-I0)/x(2))+Qmin/x(4)
    x(4)-GRmax
    -x(4)+GRmin];
Ceq=[];
end
```

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 074-0188*

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From – To)*
24-03-2011	Master's Thesis	Sept 2009 – Mar 2011

4. TITLE AND SUBTITLE
Optimization of Hybrid-Electric Propulsion Systems for Small Remotely-Piloted Aircraft

5a. CONTRACT NUMBER
5b. GRANT NUMBER
5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Rotramel, Todd A., Captain, USAF

5d. PROJECT NUMBER
5e. TASK NUMBER
5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)
Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/ENY)
2950 Hobson Way, Building 640
WPAFB, OH 45433-8865

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GAE/ENY/11-M27

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Dr. Fred Schauer (frederick.schauer@wpafb.af.mil)
Air Force Research Laboratory, 1950 Fifth Street
WPAFB, OH 45433-7251

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/RZTC

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Small electric-powered remotely-piloted aircraft (RPA) used by today's warfighters for intelligence, surveillance, and reconnaissance (ISR) missions lack desired endurance and loiter times, while the acoustics and thermal signatures of those configured with internal combustion engines (ICE) may make them unpractical for ISR. Outfitting RPA with parallel hybrid-electric propulsion systems (H-EPS) would meet the military's needs by combining the advantages of both systems while reducing fuel consumption and environmental impacts. An analysis tool was created, using constrained static optimization, to size the H-EPS components. Based on the RPA's required power and velocity for the endurance phase, an electric motor (EM) can be designed or selected and matched with a commercial off-the-shelf (COTS) propeller for maximum efficiency. The ICE is then sized for the RPA's required power and velocity for the cruise phase.

15. SUBJECT TERMS
Hybrid-Electric Propulsion System; Remotely-Piloted Aircraft

16. SECURITY CLASSIFICATION OF: a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** / **19b. TELEPHONE NUMBER** *(Include area code)*
U	U	U	UU	152	Frederick G. Harmon, Lt Col, USAF 937-255-3636x7478 (frederick.harmon@afit.edu)